Automated evaluation of generalized topographic maps

— supported by formalization and data enrichment techniques

Xiang Zhang

PhD dissertation committee

Chair		
	prof. dr. ir. A. Veldkamp	University of Twente
Promoters		
	prof. dr. M.-J. Kraak	University of Twente
	prof. dr. ir. M. Molenaar	University of Twente
Assistant promoter		
	dr. J.E. Stoter	Delft University of Technology
Members		
	prof. dr. ir. A. Stein	University of Twente
	prof. dr. F.D. van der Meer	University of Twente
	prof. dr.-ing.habil. D. Burghardt	Technical University of Dresden, Germany
	prof. dr. ir. P.J.M. van Oosterom	Delft University of Technology

ITC dissertation number 213
ITC, P.O. Box 217, 7500 AA Enschede, The Netherlands

ISBN: 978-90-6164-338-8
Printed by: ITC Printing Department

© Xiang Zhang, Enschede, The Netherlands
© Cover design by Xi Zhao
All rights reserved. No part of this publication may be reproduced without the prior written permission of the author.

UNIVERSITY OF TWENTE.

FACULTY OF GEO-INFORMATION SCIENCE AND EARTH OBSERVATION

AUTOMATED EVALUATION OF GENERALIZED TOPOGRAPHIC MAPS

— SUPPORTED BY FORMALIZATION AND DATA ENRICHMENT TECHNIQUES

DISSERTATION

to obtain
the degree of doctor at the University of Twente,
on the authority of the rector magnificus,
prof. dr. H. Brinksma,
on account of the decision of the graduation committee,
to be publicly defended
on Thursday, October 4, 2012 at 16.45

by

Xiang Zhang
born on November 26, 1982
in Hubei, China

This dissertation is approved by:

prof. dr. M.-J. Kraak (promoter)
prof. dr. ir. M. Molenaar (promoter)
dr. J.E. Stoter (assistant promoter)

*** Dedicated to my parents, wife and son ***

f

Abstract

To meet the increasing demand for geospatial information at multiple scales, and to shorten update cycles for up-to-date geo-data, national mapping agencies (NMAs) and other data providers have introduced, or are considering the use of, (semi-)automated generalization systems in their production lines. Map generalization inevitably changes certain aspects (accuracy, completeness, etc.) of spatial data. Therefore, generalized data/maps should be evaluated to see whether they are in line with the map requirements defined for a target scale and for a certain task (e.g. visualization). The use of (semi-)automated generalization systems, on the other hand, increases the need for automated quality evaluation of generalized data/maps. Automated evaluation techniques provide more objective evaluation results and reduce the time needed for the subjective visual assessment.

This thesis has identified the following issues from the state-of-the-art of automated evaluation of generalization output:

- Lack of formalized map requirements: cartographic constraints were specified at a knowledge level that is only human readable, thus not suitable for automated evaluation. For example, complex concepts, relationships, and acceptable modifications in some constraints are not well defined.

- Lack of balanced map requirements for a meaningful evaluation: previous research focused more on readability constraints and constraints for one object, or between two objects. The evaluation based on unbalanced constraints may lead to misleading (e.g. over-optimistic) results.

- Lack of explicit information/knowledge, such as characteristics, relationships and patterns in topographic data, and links between corresponding entities between different scales.

The main research question of this thesis is therefore "how to evaluate the quality of generalized general-purpose topographic data in a (semi-)automatic way, so that a meaningful evaluation can be reached?"

In the course of this research, I have made several contributions. To begin with, I have proposed an improved conceptual framework for automated evaluation of generalized maps, which consists of methodological and technological parts. The methodological part formulates empirical guidelines for selecting a set of evenly balanced constraints for a meaningful evaluation. To support automated evaluation of a set of evenly balanced constraints, e.g., readability, preservation, entity-level, property-/relation-level constraints, and constraints at individual, group and feature class levels, I have proposed a three-step approach consisting of data enrichment, data matching and evaluation. In addition, this thesis has presented a first-order representation for multi-scale data that formalizes concepts of different granularities (i.e. objects, groups and feature classes) and complex relationships (e.g. semantic and contextual ones). Cartographic constraints can be precisely defined on top of this formal theory.

Later in the design and implementation phase, I have focused on specific problems in automatically evaluating preservation constraints and the constraints on group level and feature class level. I have used building and road feature classes to demonstrate the proposed three-step evaluation approach, where settlement structures (e.g. building patterns) and spatial distributions of feature classes are extensively explored. This thesis contributes to the specific problems in the following ways:

- I have proposed an improved approach to building pattern recognition in topographic data, where different pattern types recognized from different algorithms can be combined to obtain a better recognition;

- I have proposed a generic matching approach based on pattern classification idea where various criteria can be aggregated to make decisions; besides, I have devised ad-hoc algorithms to match building patterns and alignments at different scales;

- I have proposed an approach to evaluating building alignments in generalized data, making use of the results obtained in the above steps;

- I have made attempts to measure spatial distributions of feature classes; two approaches are promising in comparing the similarity of spatial distributions: one is based on local densities and the other is based on distance.

To conclude, formalization of map specifications, data enrichment and data matching techniques form a promising framework and provide necessary instruments for automated evaluation of generalization output. This has been demonstrated by evaluating building alignments in generalized maps.

Since this research does not provide a full solution to automated evaluation, a comprehensive evaluation integrating more constraints and feature classes for a whole map is a topic for further research. This would refine and improve the current empirical guidelines for selecting a set of evenly balanced constraints for a meaningful evaluation. In addition, this research has discussed potential drawbacks of the methods proposed to address the above-mentioned specific problems and outlined ways in which they can be improved. Also this requires further research.

Samenvatting

Om tegemoet te komen aan de groeiende vraag naar geo-informatie op meerdere schalen en kortere up-date-cycli hebben topografische diensten (National Mapping Agencies) over de hele wereld (semi) automatische generalisatie geïntroduceerd in hun productielijnen. Kaartgeneralisatie verandert onvermijdelijk de kwaliteitsaspecten van de oorspronkelijke topografische data, zoals nauwkeurigheid en compleetheid. Daarom is het belangrijk om te evalueren of de gegeneraliseerde kaarten niet te veel zijn veranderd en of ze voldoen aan de kaartspecificaties op de doelschaal. Het gebruik van (semi)automatische generalisatiesystemen vraagt tegelijkertijd naar een automatische manier om deze kwaliteit te evalueren. Automatische evaluatie levert meer objectieve resultaten en reduceert de tijd die nodig is voor interactieve visuele evaluatie.

Deze dissertatie heeft als eerste de bestaande issues geïdentificeerd van state-of-the-art automatische evaluatie van gegeneraliseerde output. Dit zijn de volgende issues:

- Missende formele kaartspecificaties: cartografische constraints zijn zodanig geformuleerd dat ze begrijpelijk zijn voor mensen maar niet geschikt voor automatische evaluatie. Zo zijn bijvoorbeeld complexe concepten, relaties en acceptabele wijzigingen niet voldoende specifiek omschreven.

- Ontbrekende gebalanceerde kaartspecificaties voor een zinvolle evaluatie. Onderzoek tot nu toe heeft zich vooral gericht op constraints ten behoeve van de leesbaarheid en constraints voor één object of twee objecten. En daarmee is de set van bestaande constraints niet compleet. Evaluatie die gebaseerd is op een niet-gebalanceerde set van constraints kan tot een misleidende evaluatie leiden, bijvoorbeeld te optimistisch.

- Missende expliciete informatie en kennis die nodig is bij de evaluatie, zoals vormkenmerken, relaties en patronen van topografische

data alsook links tussen corresponderende entiteiten op verschillende schalen.

De belangrijkste onderzoeksvraag van deze dissertatie is daarmee: hoe kan de kwaliteit van gegeneraliseerde topografische data, bedoeld voor algemeen gebruik, geëvalueerd worden op een (semi)automatische manier zodat dit leidt tot een zinvolle evaluatie?

De studie die ik heb uitgevoerd om deze onderzoeksvraag te beantwoorden, heeft geleid tot een aantal wetenschappelijke bijdragen. Ten eerste heb ik een verbeterd conceptueel framework voorgesteld voor automatische evaluatie van gegeneraliseerde kaarten, welke bestaat uit een methodologisch en technisch deel. Het methodologische deel formuleert empirische richtlijnen voor het selecteren van een set van uitgebalanceerde constraints voor zinvolle evaluatie, d.w.z. constraints die rekening houden met alle relevante aspecten. Dit zijn constraints die zowel de leesbaarheid voorschrijven als het handhaven van belangrijke kenmerken ('preservation'). Daarnaast zijn de constraints niet beperkt tot individuele objecten, maar gaan deze ook over groepen en klassen. Het technische deel van de automatische evaluatie, evalueert deze constraints in een drie-stappen aanpak. Deze stappen zijn: data verrijking, data matching, en de evaluatie zelf. Tenslotte heb ik in deze dissertatie een eerste-orde representatie (theorie) gepresenteerd voor multi-schaal data welke de concepten van verschillende detailniveaus formaliseert, dat wil zeggen op object niveau, op groepsniveau en op klasseniveau, maar ook van de complexe semantische en contextuele relaties. Cartografische constraints kunnen vervolgens precies en specifiek worden gedefinieerd op basis van deze formele theorie.

In de ontwerp- en implementatie fase van mijn onderzoek, heb ik mij enerzijds gefocust op specifieke problemen van het automatisch evalueren van constraints gericht op het handhaven van bepaalde kenmerken en anderzijds op constraints voor groepen en hele klassen. Ik heb de klassen 'gebouw' en 'wegen' gebruikt om mijn drie-stappen aanpak voor automatische evaluatie, zoals hierboven beschreven te testen op bebouwingstructuren (patronen) en ruimtelijke distributie. Deze dissertatie draagt op de volgende manieren bij aan specifieke problemen rond de automatische evaluatie van gegeneraliseerde output.

- Ik heb een verbeterde methode voorgesteld voor het herkennen van bebouwingspatronen in topografische data, waarbij verschillende algoritmen worden gecombineerd voor verschillende patroontypes waardoor betere herkenning wordt bereikt. Dit heeft geresulteerd in een patroonclassificatie voor gebouwen;

- Ik heb een generieke methode voorgesteld voor het identificeren van links tussen bebouwingspatronen in verschillende data sets ('matchen') gebaseerd op de ontwikkelde patroonclassificatie. Hierbij worden verschillende criteria geaggregeerd om beslissingen te kunnen maken. Daarnaast heb ik ad-hoc algoritmen ontwikkeld om verschillende bebouwingspatronen over schalen heen te linken.

- Ik heb een methode voorgesteld om bebouwingspatronen te evalueren in gegeneraliseerde data, gebruik makend van de resultaten behaald in bovenstaande stappen.

- Ik heb testen uitgevoerd om de ruimtelijke distributie van objectklassen te meten. Twee methoden bleken veelbelovend om de gelijkheid tussen ruimtelijke distributie te vergelijken. De ene is gebaseerd op locale dichtheid en de andere op afstand.

Concluderend, het formaliseren van kaartspecificaties te samen met, technieken voor data verrijking en data matching vormen een veelbelovend framework en leveren de benodigde instrumenten voor automatische evaluatie van gegeneraliseerde topografische kaarten. Dit is aangetoond door bebouwingspatronen te evalueren in gegeneraliseerde kaarten.

Dit onderzoek levert geen complete oplossing voor automatische evaluatie. Daarom moeten zowel het integreren van verschillende constraints als het behandelen van meerdere objectklassen nader worden onderzocht. Dit zal de huidige set empirische richtlijnen voor een uitgebalanceerde set constraints voor zinvolle evaluatie verfijnen en verbeteren. Bovendien heeft dit onderzoek enkele nadelen benoemd van de voorgestelde methoden voor specifieke situaties en hoe deze kunnen worden verbeterd. Dit vraagt ook om nader onderzoek.

Acknowledgements

When I sat down in front of my laptop, and started to write down the words of gratitude, I know that my PhD career is coming to an end. Indeed, every logical research in science must have been driven by an emotional and sentimental part, from which we get the courage of our own convictions, to stand still, and to keep moving. No doubt, my PhD research and this dissertation cannot be in the present form without the help of many people.

First and foremost, I would like to express my deep gratitude to my promotor Prof. Menno-Jan Kraak, who gave me the opportunity to carry out this research at ITC. Many thanks for the stimulating discussions we had and for your kind support during my whole PhD.

Dr. Jantien Stoter, my daily supervisor, it was indeed a great pleasure and good fortune to meet and work with you for the almost five years. Any shining word is obscured in front of my sincere thanks to you. Thank you for giving me the opportunity to participate in the EuroSDR project, where I learnt a lot of my topic quickly, thank you for grounding my chaotic ideas and thoughts into perspective while leaving me a lot of free space to explore, and for providing the data to this research. I still remembered the frustrating moments in proposal and publication writing. It is your warm encouragements that guided me through and let me regain the confidence. I always enjoyed your style of making comments which always helped me to rethink about my research. Your open attitude and ways of doing concrete research silently shaped my scientific view. All I have learnt from you is a priceless treasure for the rest of my life.

My supervisor in China, Prof. Ai, Tinghua, is especially grateful for introducing me into the world of computational geometry and for sharing your knowledge in map generalization. I enjoyed our shared experiences in many projects. Also thank you for providing the code of the generalization system, without which many ideas and methods developed in this research would be still in imaginings. I am also predivided to Prof. Martien Molenaar, who has joined the supervision team since 2010. Your critical questions always dug the way directly into the core of the problems.

I would like to thank all my colleagues at the GIP department, especially Dr. Connie Blok, Dr. Rolf de By, Dr. Corné van Elzakker, Dr. Rob Lemmens, Dr. Javier Morales and Dr. Ulanbek Turdukulov for their continuous supports and encouragements. To my office-mates: Dongpo, Ioannis and Theodor for the giggling moments we had together. Thanks also to all ITC staff in general. To Loes, Ellen, Ginie, Marieke and Manon for providing logistical support; to Rebanna and Marion for dealing with financial issues.

Thanks to the people at the Department of Geography, University of Zurich, for whom I had an enjoyable and fruitful time during my stay for the EuroSDR project. Special thanks go to Prof. Dirk Burghardt, for assisting the accommodation, for kindly providing the evaluation prototype and for sharpening my understanding on automated evaluation and geospatial patterns. To Stefan Schmid, for your sharing of literatures, friendship and help in daily life. I enjoyed every moment we have worked together.

I was truely grateful to Prof. Peter van Oosterom, for our cheerful talk at the ICA generalisation workshop in Montpellier 2008, and for your recognition of the ideas in my workshop paper.

The Chinese student community at ITC is like a big family, and I will always remember the happy hours I experienced with her. Especially, several close friends must be mentioned. Dr. Tiejun Wang, your warm-hearted words and insightful thoughts influenced me deeply. I respect you for not only being a mentor in science and social life, but also as a concerned friend. Sisi, Yijian, Xuanmei, Hao Pu, Fangfang, Li Xia, Yang Ying and many others, there is no word to express my gratitude. Your admirable friendship and the countless 'Hotpot' moment are always deeply in my heart. Finally, I would like to thank Qiuju and Xiaojin for being supportive (as paranymphs) at the last moment of my PhD. I wish you all a great success and bright future.

Last but not least, I wish to express my deepest gratitude to my parents and parents-in-law, for their endless love and support, which have made life much easier. Thank you for giving me a smart brain and bringing me up to an educational level to be able to pursue science. My wife, Xi Zhao, I am so lucky to have you beside me during my stay in the Netherlands, and in the rest of my life. Thank you for filling me up with love, for being patient and supportive with me in the frustrating moments, for the joyful moments we experienced in traveling, and, of course, for bringing us our lovely son, Zhecheng. Thank you also for being my opponent when I came up with immature research ideas in the sleepless nights, which stimulated improvement and refinement. My son, Zhecheng, in the moment of writing, you already acquired many skills and were able to crawl all over the house, I wish you healthy, happy, bright, right-hearted, wise and strong.

Contents

Abstract i

Samenvatting v

Acknowledgements ix

Contents xi

1 Introduction 1
- 1.1 Background and motivation . . . 1
- 1.2 Problem definition . . . 7
- 1.3 Research objectives . . . 10
- 1.4 Methodology . . . 14
- 1.5 Thesis outline . . . 16

2 Literature review 19
- 2.1 Patterns and structures in the geospatial domain . . . 20
- 2.2 Evaluation of generalization quality . . . 25
- 2.3 Automated evaluation of generalization output . . . 31
- 2.4 EuroSDR project . . . 44
- 2.5 Summary . . . 48

3 Conceptual framework 49
- 3.1 Framework overview . . . 50
- 3.2 Analysis of map specifications . . . 52
- 3.3 Evaluation of cartographic constraints . . . 60
- 3.4 Balanced map specifications . . . 63
- 3.5 A three-step approach to automated evaluation . . . 66
- 3.6 Summary . . . 68

4 A formal theory for automated evaluation 69
- 4.1 Spatial entities and relationships . . . 70

4.2 Characterization and evaluation functions 83
4.3 Constraints as logical expressions 85
4.4 Discussion and reflections 87

5 Design of data enrichment 93
5.1 Introduction 93
5.2 Data enrichment overview 94
5.3 Patterns in building clusters 101
5.4 Recognition of building alignments 108
5.5 Measures of spatial distribution 120
5.6 Similarity between spatial distributions 127

6 Design of data matching and evaluation 137
6.1 Introduction 137
6.2 Data matching using pattern classification 139
6.3 Matching of building patterns 148
6.4 Evaluation of building alignments 156
6.5 Evaluation of spatial distribution 164

7 Implementation and validation 169
7.1 Introduction 169
7.2 Building clustering 170
7.3 Recognition of building alignments 172
7.4 Data matching via supervised classification 183
7.5 Preservation of building alignments 191
7.6 Preservation of spatial distribution 200

8 Conclusions and recommendations 211
8.1 Conclusions 212
8.2 Contributions 225
8.3 Recommendations for future work 229

A Auxiliary geometric structures 233
A.1 Delaunay triangulations 233
A.2 A Voronoi-like diagram based on skeleton 237

B Expert survey 243

Bibliography 251

Author's Biography 271

1 Introduction

1.1 Background and Motivation

Nowadays, geospatial information at different scales is highly demanded for different types of information consumers, ranging from the general public to government sectors and from industrial applications to scientific inquiries. To meet the increasing demand and to shorten update cycles of smaller scale maps from larger scale maps, national mapping agencies (NMAs) and other data providers have introduced, or are considering the use of, (semi-)automated generalization systems in their production lines. As a result of the changing production lines, up-to-date geospatial data are now published in all kinds of mediums, from paper maps to web-based map portals, or to digital maps displayed on hand-held mobile devices such as smart phones. It is therefore important, may be more important than ever, for data suppliers to provide high quality multi-scale geo-information.

On the one hand, the quality of (general aspects of) spatial data had already been studied for decades [77, 79, 176] which also resulted in a large number of standards such as USA-SDTS, CEN/TC287 and ISO/TC211 [208]. On the other hand, only a few work was done for assessing the quality of spatial data obtained via map generalization [e.g. 19, 98]. Map generalization defines quality criteria which are slightly different from general quality standards for spatial data, and also requires a different way of evaluation [120]. Traditionally, the quality of generalized maps is evaluated by cartographic experts working with visual methods, which is both subjective and time-consuming. Automated evaluation techniques are thereby required in such a context to provide more objective evaluation results and to reduce the time needed for the subjective visual assessment.

The following sections further explain several key concepts and aspects related to the research topic of this thesis. First, different views on the

quality of generalization output[1] and the view adopted in this research is described in Section 1.1.1, where cartographic constraints that are used to define quality criteria (map requirements) are also introduced. Second, we distinguish between general-purpose and task-dependent generalization for the purpose of clarifying our focus on the evaluation of generalized general-purpose topographic data/maps (Section 1.1.2). Third, the importance of geographic patterns and structures in relation to generalization and quality evaluation is briefly discussed in Section 1.1.3, since patterns are group level structures which will be one of the focus of this thesis.

1.1.1 *Generalization quality and cartographic constraints*

Although recent automated generalization techniques were improved considerably [e.g. 187, 193], the quality of generalization output is still not fully satisfactory [192, 193]. This doubles the need to explicitly evaluate generalization quality and convey the quality information to intended users.

Quality is apparently a vital issue to map generalization, because map generalization inevitably changes or modifies certain aspects (e.g. position, attribute and completeness) of the map/data. This in general reduces the data quality, especially in the context of automated generalization.

There are two perspectives in which the changes introduced by map generalization can be viewed. The first one regards the changes as inaccuracies or errors. It means that when a change occurs it definitely reduces the quality of spatial data. This view is in line with the definition of spatial data quality in general, i.e., positional accuracy, attribute accuracy, completeness, consistency, etc. João [98] for example carried out a study and found that generalized maps are prone to errors. She analyzed paper map series and identified that qualities such as positional accuracy and lengths of objects decrease as map scale becomes smaller. However, this seems to be not a real problem because to generalize maps the above qualities are bound to decrease. Additionally, this type of quality analysis is not able to answer questions like to what extent is the generalized map useful. Map users also have no idea what does it mean and how should they do with it.

Another perspective equates the generalization quality with the fulfillment of map requirements or specifications for specific tasks (a *fitness for use* concept), which is perhaps more appropriate in the generalization domain. In it, the introduced changes are the result of fundamental cartographic

[1] By *generalization output* we mean generalized maps or generalization solutions in vector data format.

conflicts. When representing a larger scale map at a smaller scale, conflicts arise between the increasing amount of features and symbol weights to be represented and the limited map space to represent them. To resolve the conflicts modifications to the map are unavoidable. Nevertheless, as long as such modifications do not reduce or if they even improve the fitness of the map for a given task, e.g., visualization, the generalization is regarded as an acceptable or successful solution.

To generalize spatial data for visualization (i.e. cartographic generalization), for instance, some features have to be displaced away from their original locations (thus decreasing the positional accuracy), or otherwise the features will be visually cluttered, leading to a bad visualization. As long as the decrease of positional accuracy does not change e.g. relative positions among objects, an intentional decrease in positional accuracy (within an allowed threshold) is acceptable. This way, the generalized map could enable users to better perceive, understand, and communicate the spatial information (readability improved) without changing important relationships. While in navigation-related applications, the positional accuracy of map features, especially roads and landmarks, is regarded to be a more important issue. To summarize, map generalization in this sense is an optimization between conflicting requirements, and the generalization quality depends on the specific task at hand.

As a result, a key question to the evaluation is how can one define a successful generalization? Or more specifically, whether or to what extent are the modifications acceptable for a specific generalization task? And eventually, how can one formally define a generalization task?

In general terms, a successful generalization can be defined in many different ways, e.g., suppressing unnecessary details to discern important patterns, enabling a user to succeed in a given task (e.g. exploring, way finding, and navigation), ensuring a visually balanced and pleasant presentation and so on [120]. The major difficulty of generalizing datasets/maps and evaluating their qualities is to specify these types of requirements in a format and knowledge level that is both human and machine readable. By addressing this difficulty the systems should be able to generate higher quality smaller scale maps from detailed maps automatically or semi-automatically.

As a recent common practice, the generalization community nowadays focuses on the use of *cartographic constraints*, or shortly *constraints*, to de-

fine map requirements and to control and evaluate automated generalization [19, 22, 23, 85, 88, 89, 131, 144, 172, 215]. A constraint is a statement that specifies how generalization output should look like. In other words, a constraint defines target states of map features and their properties after generalization, and/or in relation to their initial states. For example, 'sizes of buildings should be larger than X mm^2', or 'target shape should be similar to initial shape'. Cartographic constraints are declarative in nature, which means they separate the goal from the way to achieve it, thus ensuring a more flexible modeling. Hence we naturally use constraints in this thesis to express map requirements for automated evaluation. In general, cartographic generalization defines two fundamental requirement, namely, *readability constraints* which improve the readability of the map and *preservation constraint* which keep important geographic characteristics, relationships, and patterns of the map.

The use of the term resembles the way how constraints are used in *constraint programming* in computer science and as *integrity constraints* in database domain. The meaning of the term in the generalization domain is however slightly different (see Section 2.3.2 for details).

1.1.2 *General-purpose and task-dependent generalization*

Topographic data/maps, also called general-purpose or reference maps, are currently the main product of most NMAs [192]. Many different but similar definitions of topographic maps can be found in the literature, for example the following:

> "Topographic maps supply a general image of the earth's surface: roads, rivers, buildings, often the nature of the vegetation, the relief and the names of the various mapped objects." [104, p. 44]

From the production perspective, one can distinguish between general-purpose data production and task-dependent data generation [188, 189]. At an institutional level (e.g. NMAs), topographic data are generalized and maintained at multiple predefined scales. They are designed to be as general as possible, usually showing features such as roads and railways, buildings, land use, hydrography, infrastructure, and administrative boundaries. As a result, they can be used for GIS analysis (if correct data structure such network topology is kept), orientation, displayed as paper and digital raster maps (e.g. relief and city maps), as base maps for thematic mapping as well as for applications in different domains and for different tasks [e.g. 193].

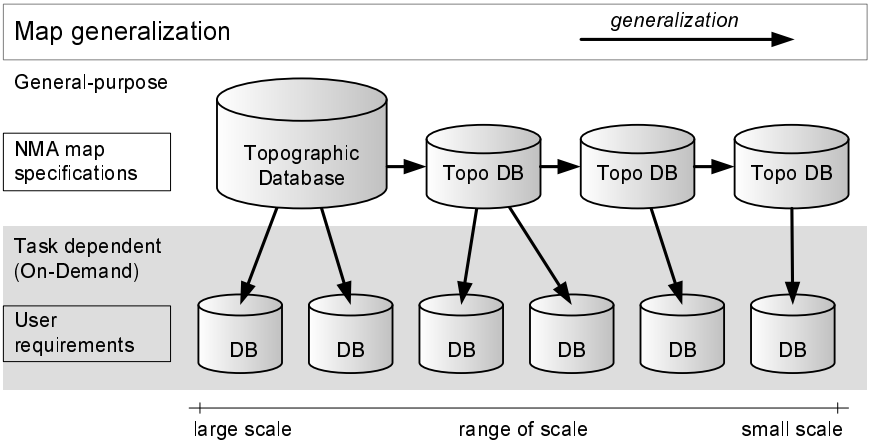


Figure 1.1 General-purpose topographic data and task dependent data generation (on-demand mapping): concepts extended from the NMA generalization models described in [188, 189]

Typically, further processing (e.g. symbolization and/or generalization) is required to generate data for various tasks from general-purpose topographic data (Fig. 1.1). Note that in describing general-purpose generalization, we do not distinguish model generalization from cartographic generalization, in order to emphasize its relation to task-dependent generalization. This is also due to a lack of well-defined boundary between the two in the production domain [see e.g. 189]. Task-dependent generalization is also referred to as *on-demand mapping* because it generates data/maps according to user requirements on demand [see e.g. 49, 68].

The major differences between general-purpose and on-demand generalization are summarized in Table 1.1. This research focuses on evaluating the quality of generalized general-purpose topographic data/maps, where NMA map specifications are the main source to define the quality of generalization output. This is justified by the fact that topographic map production is still dominant and important in most NMAs [189, 192].

User requirements for on-demand mapping, on the other hand, are out of the scope of this research. Nevertheless it is worth noting that, NMA specifications are more generic (i.e., task-independent) than user requirements. In other words, NMA specifications define fundamental cartographic requirements: a map that is readable (*readability constraint*) and that

	General purpose	*Task dependent (on-demand)*
Theme	*all basic feature classes*	*specific feature classes are emphasized, while some other classes may be removed*
Scale	*predefined scales*	*arbitrary scales, which may not be defined within NMAs*
Map requirements	*NMA specifications*	*user requirements, which are sometimes hard to be translated in cartographic terms*
Degrees of LOD	*more or less the same LOD*	*LOD varies between regions of interest, or between feature classes, depending on the task*

Table 1.1 Major differences between the two generalization modes

preserves geographic characteristics, relationships, and patterns (*preservation constraint*). User requirements can be generally characterized by the accessibility, relevancy, completeness, timeliness, interpretability, the ease of understanding, and the costs of the data [134]. It is evident that user requirements are based on the fundamental cartographic requirements. Base maps should first satisfy the fundamental constraints as much as possible, before user requirements can be further addressed for on-demand mapping (Fig. 1.1).

1.1.3 *Characteristics, relationships, and patterns*

Cartography not only serves as a tool to communicate spatial information to the general public, but also provides powerful means for geographical analysis. For example, when we look at a map, we do not only see individual objects, but also we see characteristics, relationships, and patterns among objects or groups of objects, which in turn gives insight into the underlying processes or socio-cultural/historical functions of the geographic phenomena which generate the patterns (i.e. a data exploration and pattern confirmation process) [116, 119].

Map generalization should discern important patterns and structures and allow the recognition of pattern generating processes at smaller scales [135, 136]. The principle is in general the same whether is it for topographic or thematic mapping. In thematic mapping (e.g. land use mapping), the distribution pattern of a statistical surface (e.g. land use types on the map) is considered highly important and should be preserved [135, p. 460]. We

will give more examples of geospatial patterns for topographic mapping in Section 2.1.

As also noticed by Bard and Ruas [21], *"generalization is not a mere reduction of information – the challenge is one of preserving the geographic meaning"*, and thus the focus of evaluation of generalization output should be on the quality of the abstraction. Abstraction means, on the one hand, to simplify the representation by getting away from unnecessary details, and on the other, to enhance important characteristics, relationships and patterns given a specific purpose. Therefore, the characteristics, relationships, and patterns should be explicitly measured and evaluated to meet the above functions of generalization [119]. Also, it has to be mentioned that depending the theme and scale, a particular subset of all patterns may appear on the map [119].

In this thesis, the proposed evaluation approach will focus especially on the preservation of characteristics, relationships, and patterns. This has hardly been studied before because firstly related constraints were difficult to formalize [44, 192], and secondly extra effort is needed to enrich existing data for evaluating the characteristics, relationships, and patterns. In conventional manual generalization, such important characteristics, relationships, and patterns are visually identified and well respected by cartographers. For automated processes, they should be firstly explicitly identified (data enrichment) in the database and then evaluated with respect to their initial states (formal map specifications).

1.2 Problem definition

In general, we take a bottom-up approach to the evaluation of generalization output. That is, instead of using an overall quality criterion for a whole map, as usually the case in expert visual evaluation, we firstly decompose the generalization quality into various aspects and then assess the different aspects individually. Afterwards the individual assessments could be aggregated into a single decision [19, 21, 120], or otherwise be visualized [144], to indicate the overall quality of the generalization output. The major difficulty of using a single quality criterion in a computerized evaluation is that it is too difficult (if not impossible) to formalize the criterion. For example, how to formalize degrees of generalization and the requirement for an aesthetically pleasing map [219]? On the other hand, while it is also difficult to aggregate after the decomposition, this bottom-up approach seems to be the only way to automate the evaluation process because the decomposed aspects can be formalized and computed. In addition, the

bottom-up approach is able to give insights into different quality aspects of the data that are still meaningful to intended users.

The first issue in this process is to define quality criteria, i.e., map requirements for generalized datasets, by which the success of the generalization can be evaluated. To automate the evaluation, then, map requirements and the processes needed for the evaluation should be formalized. As many kinds of information and knowledge are not encoded in the data, data enrichment techniques are required to make them explicitly available to automated processes. The final issue is converting the assessments of individual requirements in a form that could reflect the quality of the whole map, and that is meaningful to intended users (e.g. cartographers).

In short, formalization is a key concept for automation, which is needed in our research at specification level (constraint formalization), data level (data enrichment) and process level (process formalization). The problems that this research mainly focuses on are defined as follows.

1.2.1 *Lack of formalized map requirements*

The cartographic constraints that will be studied in this thesis are not a new proposal. In fact, they are based on previously collected knowledge and constraints for automated generalization and evaluation of topographic data/maps [23, 190, 192]. Specifically, the constraints in this research are derived from NMA map specifications which impose basic requirements for general-purpose map generalization [192].

However, many previously formulated constraints are specified at a knowledge level that is only human readable. Thus, the constraints need to be formalized to contain necessary information to be able to steer automated evaluation of generalization output. For example, map features and their properties (i.e. characteristics, relations and patterns) in the constraints, measures to the constrained properties, and acceptable modifications to the properties should be well defined.

1.2.2 *Lack of balanced map requirements for a meaningful evaluation*

From a technical point of view, it is feasible to realize an automated evaluation system based on currently developed measures and evaluation approaches. However, the issues that impede the development of a meaningful evaluation system is not only technical but also methodological. Previous studies on the evaluation of generalization output were primarily based

on the constraints that are mostly easier to implement such as size, shape, position of individual objects [42, 44, 45]. The use of a set of unbalanced constraints may cause problems in interpreting the evaluation results, leading e.g. to an over-optimistic assessment [88]. An over-optimistic result might be obtained if the dataset is evaluated against a set of constraints that is also used to guide the generalization process. The evaluation based on a set of unbalanced constraints can be misleading and hence is less meaningful.

To address this issue, a complete or at least a set of balanced constraints is required. In fact, we never lack a comprehensive list of constraints in the literature, see for example the AGENT project [1], the EuroSDR project [190, 192] and books on automated generalization [e.g. 121]. The key problem is that a set of balanced constraints were previously not included in evaluation systems in a meaningful way, which makes the automated evaluation subject to the above-mentioned bias or over-optimism. This thesis intends to focus on different levels of analysis which give a more complete view of the quality of generalized topographic datasets. As a proof-of-concept, this research will evaluate group level (e.g. alignments) and feature class level (spatial distribution) preservation constraints, which are hard to formalize due to lack of information. An example in Section 3.4 illustrates the idea of evaluations on balanced and unbalanced map specifications, and motivates our focus on the constraints that preserve the characteristics of spatial phenomena at different levels (especially the group and global level).

1.2.3 *Lack of explicit knowledge in multi-scale databases*[2]

The problems described above can be categorized as specification level problems. To enable automated evaluation, on the other hand, various kinds of information are required at data level as well. For example, it is impossible to automatically evaluate density related constraints if the density information is not available. Although geometries (e.g., coordinates) and basic attributes are encoded in common data models and are thus accessible to the evaluation system, many more are not explicitly available and hence need to be added, e.g., from other sources or, via data enrichment techniques.

A related issue is that the use of different levels of analysis naturally requires that computers can handle higher order phenomena, semantics and

[2]The term *multi-scale database* is used to indicate the need for explicit inter-scale corresponding relationship; the term does not limit its usage to databases but includes data or datasets as well.

concepts such as *pattern* and *spatial distribution* in a meaningful way. Such knowledge, which is traditionally interpreted by cartographers from generalization situations, is commonly not encoded in topographic datasets. As a result, topographic data should be enriched to make the knowledge available to the systems.

Another type of implicit information is the links between corresponding entities at different scales that represent the same real-world phenomena. This inter-scale relationship is indispensable for multi-scale databases for maintaining consistency and propagating update across scales. Automated evaluation, in particular the evaluation of preservation constraints, also relies heavily on the relationship [42, 44, 192]. The links between corresponding objects are not commonly available, and therefore they should be enriched between datasets at multiple scales.

1.3 Research Objectives

The main objective of this thesis is to answer the question as to how to evaluate the quality of generalized general-purpose topographic data in a (semi-)automatical way, so that a meaningful evaluation can be reached. A meaningful evaluation is briefly defined in Section 1.2 and can be divided into two parts. The first part concerns map specifications and related questions are:

1. *What are the map specifications that define the quality of general-purpose map generalization and how should different map specifications be evaluated?*
2. *What are the problems in the selection of map specifications for automated evaluation and what are the principles in selecting the specifications for a meaningful evaluation?*
3. *How to formalize map specifications so that quality criteria for automated evaluation can be unambiguously expressed? In particular, how to formalize complex concepts (e.g. networks, alignments, and patterns) and relationships (e.g. connectivity, density, and spatial distribution) prescribed in the specifications? Which formal concepts are useful in the formalization process?*
4. *To what extent does knowledge obtained from experts or interactively (manually) generalized data help to formalize map specifications?*

The second part concerns key technical aspects and system components (see also Section 1.2) towards a meaningful automated evaluation system. Related questions are listed below:

5. *What are the requirements (or components) for an automated evaluation framework to be meaningful?*

6. *Which types of information are required by automated evaluation but are not encoded in topographic data to be evaluated and how can the information be added to current data and evaluated?*

7. *How can the key concepts and components in the proposed evaluation framework be validated and to what extent does knowledge obtained from experts or interactively generalized data help?*

8. *What are the capabilities and limitations of the proposed framework for automated evaluation?*

In the following, Section 1.3.1 defines the topics within the scope of this research, as well as the topics that are related but are out of the scope of this research. Section 1.3.2 describes potential application domains and parties that might benefit from this thesis work.

1.3.1 *Research scope*

First, this research focuses on automated evaluation of generalized general-purpose topographic datasets, and therefore on the map specifications of NMAs (see Section 1.1.2 for the distinction between general-purpose and task-dependent (on-demand) generalization). A survey carried out by NMAs among their customers shows a continuous demand for traditional, paper maps representing topography at various scales [192]. This implies that NMAs still have to produce multi-scale topographic datasets, and justifies our focus. The evaluation of on-demand mapping, i.e., the study of user requirements for task-dependent generalization, is out of the scope. Note however that the outcome of this research is still highly relevant for on-demand (web) mapping.

Second, this research focuses on large- to mid-scale generalization, which is considered as the most time-consuming generalization task of current production lines [190, 192]. Besides, we mainly concentrate on building and road feature classes in this thesis due to their relative importance and complexity in the generalization within the focused scale range. This focus is also in line with the findings that NMAs generally imposed more constraints on buildings and roads than other features [44], and with the extent of scientific research devoted in the generalization of buildings [e.g. 17, 41, 53, 94, 154, 156, 165, 172, 214, 215]. In such a scale range, settlement structures are one of the most significant features on the map, and hence

the characteristics, relationships, patterns, and spatial distributions thereof should be retained or emphasized [184]. Except for connectivity, density and spatial distribution, group properties of networks such as structures and arrangement, on the contrary, are mostly not visible in such a scale range. As a result, we will concentrate on the quality of settlement structures as a proof-of-concept.

This research deals with both readability and preservation constraints, as it was initially motivated by cartographic generalization. Nonetheless we do not explicitly distinguish model generalization (i.e. DLM — digital landscape model) from cartographic generalization (i.e. DCM — Digital Cartographic Model). This is mainly because many NMAs do not distinguish between the two, or the two models are used in a mixed manner, or it is still an open issue to be decided [70, 189]. Kadaster, in particular, integrates aspects of DLM and DCM, i.e., the geometries in the datasets take into account the way features are visualized on the map [193, 194]. One should also note that this research focuses more on issues in the evaluation of preservation constraints, especially the constraints on group and feature class level. This means that the findings of this research are also beneficial for the generalization of DLM (i.e. model generalization) which strives to preserve initial representations.

Previously, the generalization process (e.g. measures, algorithms and their parameters) were evaluated in the context of European Organization for Experimental Photogrammetric Research (OEEPE; the predecessor of EuroSDR) [164]. This is out of the scope of this research. Also, we will not generalize topographic data using self-developed generalization operations. Instead, the datasets used in our research (mainly from Kadaster) are generalized either interactively by cartographers or (semi-)automatically by generalization systems. In short, the focus is on the evaluation of generalized data but not the generalization process. Nevertheless, the outcome of this research is highly relevant to automated generalization and multi-scale database research.

This research does not intend to develop a functioning automated evaluation system, as a fully automated evaluation system is considered not feasible at this stage. Instead, this thesis attempts firstly at the definition of a conceptual framework for automated evaluation which can bridge the gaps in the previous research on automated evaluation of generalized data, and then the validation of the concepts in the framework by prototyping key aspects and system components.

The key aspects and system components of a meaningful evaluation framework include (i) formalization of map specifications (constraints), (ii) automated evaluation framework and its formalization, (iii) data enrichment of multi-scale topographic data that enriches implicit entities and properties of features out of topographic datasets, (iv) data matching that establishes the links between identical objects of different levels of detail, and (v) aggregation of individual assessments into a single value, providing a view of the quality at a higher level.

When we match the multi-scale data for evaluation, the spatial component is a major concern, though the schema matching at a conceptual level that links semantically similar or related classes together is also important and interesting. In our research the datasets of different scales to be evaluated are maintained in the same institution (i.e. within the same NMAs), meaning that the semantic relationships between classes are well-defined. Therefore schema matching for integrating heterogeneous database schema is out of the scope.

1.3.2 *Relevance*

Automated evaluation of generalization output is useful by itself but it can also be used to improve the generalization systems [164]. The evaluation has the following merits. First, it helps to foster better knowledge on requirements (i.e. specifications) for map generalization, by which one can explicitly define what a successful generalization is. Meanwhile, the knowledge on map specifications can be used to improve the generalization systems, since well-defined specifications formalize the objectives of the generalization for automated processes. Second, the evaluation can identify cartographic conflicts and thereby trigger generalization operations for solving the conflicts. This can be beneficial in both interactive map editing (evaluation for editing) and systems with self-evaluation capabilities, e.g., those based on optimization or (multi-)agent techniques (evaluation for automatic triggering and parametrization). Although such self-evaluation generalization systems are not the aim of this thesis, they can also benefit from the outcome reported in this thesis. Third, the evaluation results can be used to compare different generalization solutions, e.g., to identify acceptable ones or poorly generalized ones, with respect to specifications.

On the other hand, our attempt at automated evaluation processes have added values. First of all, automated evaluation improves the objectivity of the quality assessment. Second, the evaluation will inform users of where

poor results occur and how they could possibly respond, e.g., by indicating where poor generalization occurred or proposing operations to improve the current solution. Third, automated evaluation can be integrated into generalization systems to improve the quality of automated solutions.

1.4 METHODOLOGY

This section describes the methodology to address the research objectives and questions described in Section 1.3. It also provides information on tools and test datasets used for implementation and validation.

1.4.1 *Research design*

First, literature review will be carried out to identify the state-of-the-art of automated evaluation of generalized data. The focus is on the one hand on previous studies of map specifications based on which the generalization quality was evaluated, and on the other hand, on technical aspects of how previous evaluation approaches are designed. The literature study will identify the gaps in previous research and the needs for an improved evaluation framework (e.g. needs for data enrichment).

To bridge the gaps, an improved conceptual framework will be designed based on the findings in the literature review. Among other aspects, map specifications (constraints) are the most pertinent to the definition of an evaluation process, depend on which choices can be made as to which type of information should be enriched, which objects should be matched, etc. To unambiguously define cartographic constraints, a formal theory (in a first-order representation) will be developed where firstly a simplified Entity Relationship model will be defined for automated evaluation and then cartographic constraints will be formalized on top of the ER model. The formal theory only provides a formal syntax (structure) with which the constraints can be clearly defined and unambiguously understood. Further formalization should address parametrization issues such as what are the acceptable changes (thresholds and tolerances) at scale transitions.

The next step is the logical design which serves as a translation between the conceptual framework (theory) and the implementation (practice). In this phase, key aspects and framework components will be designed with respect to the focused feature classes such as buildings and roads. Based on our conceptual framework, the logical design can be divided into two subcomponents, namely the design of data enrichment and the design of data matching and evaluation. Methods and techniques in computational

geometry, pattern recognition and machine learning will be extensively applied and extended in the design with their potentials explored.

Next, the evaluation framework is implemented by prototyping key aspects and system components described in the logical design phase. Experiments will be carried out to show the potentials of the concepts and designed components. On the other hand, existing interactively generalized data/maps based on both visual and statistical analysis will be studied to get insight into the formalization of map specifications (e.g. determine the acceptable changes of constrained properties). Several individual constraints at the group and feature class level will be evaluated with respect to the topographic data as described in Section 1.3.1.

Finally, validation of the evaluation framework and some of the key concepts and system components helps to identify the possibilities and limitations of the proposed framework. In this respect, cartographic experts have been invited to validate part of the implementation. In addition, interactively generalized data as another way to validate whether or not the evaluation results are appropriate will be used.

1.4.2 *Tools and test datasets*

The key components and functionalities were implemented in an interactive generalization system called DoMap. Besides machine learning algorithms that are integrated from third party software (and packages), the implementations are based on self-developed data structures and algorithms written as extensions to DoMap. For a more detailed account for the tools used and implementations, the readers are referred to Chapter 7.

The topographic datasets used to validate the proposed framework and components come partly from the EuroSDR project on the state-of-the-art of automated generalization in commercial software [190] and partly from Kadaster (NMA of the Netherlands). The source (initial) datasets are ungeneralized topographic data at 1:10k and 1:25k (see details in Section 2.4.1). The target datasets are interactively, manually generalized data (from Kadater) and automatically generalized solutions (from the EuroSDR project), both at the scale of 1:50k. The constraint set inspected in this thesis is derived mainly from the generic set of constraints defined by the experts of NMAs across Europe within the context of the EuroSDR project [190, 192].

1.5 THESIS OUTLINE

This section gives an outline of the structure of the thesis.

Chapter 1 (this chapter) gives a general introduction to the research topic of this thesis. It firstly introduce the general background which motivate the research on automated evaluation and clarifies concepts for the proposed research. Then problems are summarized based on which research objectives and questions are formulated. To answer the research questions, research methodology, tools and test data used are explained.

Chapter 2 reviews literature related to the proposed research topic. Since the focus of the evaluation is on groups (meso-level entity) and feature classes (macro-level entity), typical patterns and structures in the geospatial domain are identified and reviewed in relation to map generalization. Furthermore, previous efforts on approaches and techniques to quality evaluation in generalization are reviewed which reveals needs for further development in automated evaluation of generalization output.

Chapter 3 proposes an improved framework for automated evaluation of generalization output, in order to address the issues described in Section 1.2. The framework is driven by formal map specifications and consists of data enrichment, data matching, and evaluation components. Hence aspects of map specifications (cartographic constraints) and how different types of constraints should be evaluated are analyzed. This analysis provides a basis for the formalism presented in Chapter 4. Next to the formal constraints is the requirement to balance between various constraints in order to arrive at a meaningful evaluation. Another important part of the framework that actually evaluate the generalized data is formulated as a three-step model (i.e. to enrich, to match, and to evaluate data).

Chapter 4 presents a formal theory in which map specifications (cartographic constraints) can be unambiguously expressed in a first-order predicate representation. The theory explicitly defines three levels of entities, namely individuals (micro-level), groups of objects (meso-level), and feature classes (macro-level), and formally describes the (spatial and non-spatial) relationships between these entities. Groups examples such as networks and alignments of objects are formally defined for the sake of this research. Finally, map specifications of

different degrees of complexity are formalized based on the entity-relationship model, with the expressiveness of the formalism discussed.

Chapter 5 describes the design of data enrichment component as applied to building (and in some accessions network) features as a proof-of-concept. It starts with introductions to low-level data enrichment operations (i.e. measures on individuals), proceeds with automated enrichment of building patterns and alignments (recognition of pattern instances and measure of their characteristics), and ends with attempts to measure spatial distributions of feature classes and compare their similarities.

Chapter 6 presents the design of data matching and evaluation components, with a focus on group and feature class level constraints for building (and in some occasions network) features. Two data matching approaches are described. The first one matches spatial objects of any type using a pattern classification framework. The second one matches only alignments of objects using a specifically designed algorithm. After that, automated evaluation of building alignments and spatial distribution (based on local density measures) are explained.

Chapter 7 implements the concepts and components described in Chapters 5 and 6, validates the feasibility of the approaches using expert knowledge and interactively generalized data, and discusses the limitations of each approach.

Chapter 8 summarizes the findings of this research as well as insights into a meaningful framework for automated evaluation. It first concludes the whole thesis by answering the research questions. Then it highlights main contributions to the research fields. Finally it closes the thesis by outlining recommendations for future work.

Appendix A introduces technical details of frequently used auxiliary data structures such as Delaunay triangulations and a Voronoi-like diagram.

Appendix B is a survey template aiming at collecting expert knowledge concerning the generalization of building patterns.

2 Literature review

This chapter reviews literatures related to the topic of automated evaluation of generalized general-purpose topographic data/maps. Since one of the subject matter of the evaluation is the geospatial patterns (i.e. characteristics and relationships of groups of spatial objects), an overview of the patterns and structures in the geospatial domain is given in the first place (Section 2.1). As map generalization provides insight into the underlying pattern generating process [135], it is crucial to keep or enhance some of the patterns for specific map purposes and scales [184].

In the second part of this review, previous effort on the quality assessment for generalization is described. To begin with, Section 2.2 gives an overview of previous approaches to assessing the generalization quality, in which different types of quality evaluation for map generalization is introduced. After that, Section 2.3 narrows down to the topic of automated evaluation of generalization output and previous efforts in addressing some of the problems. Then, the EuroSDR project on the state-of-the-art of automated generalization is shortly described (Section 2.4), due to its close relation to automated evaluation techniques and especially to this thesis work. Finally, remaining issues are summarized in Section 2.5.

Own publications

This chapter is partly based on the following publications:

- X. Zhang, J. Stoter, and T. Ai. Formalization and automatic interpretation of map requirements. In *The 11th ICA Workshop on Generalisation and Multiple Representation*, page 12, Montpellier, France, 2008.

- X. Zhang et al. Pattern classification approaches to matching building polygons at multiple scales. In *(ISPRS Congress 2012)*, Melbourne, 2012 (to appear).

- X. Zhang, T. Ai, J. Stoter, M.-J. Kraak, and M. Molenaar. Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. *GeoInformatica*, page 33 (online first). doi: 10.1007/s10707-011-0146-3.

2.1 Patterns and structures in the geospatial domain

Patterns that exist in our geographic world should be portrayed on maps depending on map purposes and scales. The following quotation from the USGS education tips[1] shows the relationship among geographic phenomena, topographic maps and, scales of the representation:

> "Examine a list of physical features such as eskers, drumlins, glaciers, bajadas, canyons, mesas, coastal features, sand dunes, marshes, floodplains, volcanoes, alluvial fans, karst, and so on. How are these features identifiable on topographic maps? What are the sizes of each of these features? What is the most suitable scale with which to examine the entire physical feature? How do physical features affect population settlement amount and patterns?"

From this we know that, there are various kinds of geographic phenomena (i.e. higher-level concepts and patterns) in the real world that need to be visually represented, and each of the phenomena have a different size and thus an appropriate scale should be determined for its optimal representation.

This section provides an overview of the patterns and structures in the geospatial domain. To begin with, Section 2.1.1 explains the use of the term pattern in different geospatial contexts and discusses their relationship to generalization terms. Section 2.1.2 gives several typical examples of geospatial patterns that are commonly encountered in map generalization domain. Section 2.1.3 discusses the relationship between patterns, generalization, and scale.

2.1.1 *Meanings of geospatial patterns*

The term pattern/structure in the geospatial domain is loosely defined. There are a lot of synonymous terms that can be found in the literature of the geospatial domain: higher-level concept and semantics, pattern, structure, network, spatial distribution, spatial arrangement, spatial layout, etc.

[1] `http://rmmcweb.cr.usgs.gov/outreach/topoteach.html`

These terms cannot be used to categorize the high-level concept, because they are neither well defined nor mutually exhaustive. There are some example definitions of these terms: AGENT [2] adopted the definition of pattern from Wentz [223] as the organization of phenomena in geographic space that has taken on specific regularity, which in turn is taken as a sign of working of a regular process. Weibel and Dutton [220] described structures as complex concepts that involve relationships that are often not explicitly coded but must be inferred by analysis. Sadahiro [168] studied spatial distribution of points and defined spatial arrangement as an indication of location of points with respect to each other, which is invariable with rotation, translation and scaling. Mackaness and Edwards [119] introduced a common definition of pattern: it is a property within an object, or between objects that is repeated with sufficient regularity. Steiniger [185] distinguished further two types of pattern. One is visual pattern which subjects to certain perceptual organizations like Gestalt principle discovered by Wertheimer [225] and is visible to everyone. The other is domain-specific pattern which are only accessible to person with appropriate domain knowledge. The definitions of the terms are not clear enough for automation process, but they still provide enough information for us to understand the concept of pattern.

To unify the use of the term, we distinguish between *generic pattern* and *domain-specific pattern*. The generic pattern describes the spatial distributions of (multiple) variables over geographic spaces, their exceptional values, spatiotemporal dynamics, and associations. It is usually the output of a spatial data mining process which aims at discovering interesting and previously unknown, but potentially useful knowledge from large datasets (e.g. determine the 'hot spots' in crime analysis) [175]. Because the discovery and description method for such patterns are generic (e.g. those for spatial outliers, co-locations and clustering), we term them generic patterns. To characterize the spatial distribution of a set of objects is to describe the generic pattern of these objects. The domain-specific (geo-spatial) patterns, on the other hand, have four fundamental features:

1. The detection of such patterns does not yield any new knowledge. In fact, most of them are results of geographic processes that are already known;

2. Most of such patterns are recognized and described in *ad-hoc* manners (e.g. those for building, river, and street patterns);

3. Such patterns have typical structures and the rules to describe their spatial organizations may quite differ from each other;

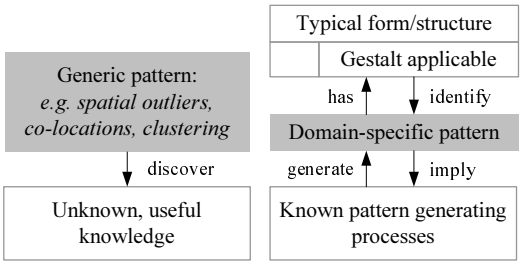


Figure 2.1 Differences between two kinds of patterns in the geospatial domain

4. Underlying geographic processes which generate different patterns can be inferred from the structures of the patterns.

The differences between generic and domain-specific patterns are shown in Fig. 2.1. It is noticeable that although domain-specific patterns have their distinctive spatial arrangements, only some of them can be explained by Gestalt principles of perceptual organization [225]. This is in line with the visual pattern as proposed by Steiniger [185]. For the patterns whose arrangements can be explained by Gestalt principles (e.g. clusters and alignments), the principles provide rules to recognize the patterns from topographic data. For the pattern that cannot (or not completely) be explained by the principles (e.g. soil erosion patterns), different detection approaches have to be devised. In the next section, we introduce several typical domain-specific patterns that are common in map generalization.

2.1.2 *Examples of geospatial patterns*

This section presents some typical geographic patterns. For each of them, we begin with a geographical perspective description and end up with a discussion in map generalization domain.

Drainage patterns

The drainage pattern development should be put into the context of river morphology and channel process [126], in order to better understand them. There exists a close relationship between local topographic and geologic factors and the development of drainage patterns, as e.g. described by Ritter [159]:

> "Over time, a stream system achieves a particular drainage pattern to its network of stream channels and tributaries as determined by local geologic factors. Drainage patterns or nets are classified on the basis of their form and texture. Their shape

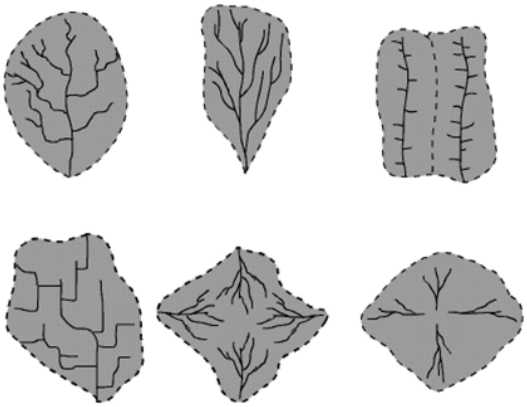


Figure 2.2 Examples of drainage patterns (from left to right): dendritic, parallel, trellis (top), rectangular, radial and centripetal patterns (bottom)

> or pattern develops in response to the local topography and subsurface geology. Drainage channels develop where surface runoff is enhanced and earth materials provide the least resistance to erosion. The texture is governed by soil infiltration, and the volume of water available in a given period of time to enter the surface. If the soil has only a moderate infiltration capacity and a small amount of precipitation strikes the surface over a given period of time, the water will likely soak in rather than evaporate away. If a large amount of water strikes the surface then more water will evaporate, soaks into the surface, or ponds on level ground. On sloping surfaces this excess water will runoff. Fewer drainage channels will develop where the surface is flat and the soil infiltration is high because the water will soak into the surface. The fewer number of channels, the coarser will be the drainage pattern."

A drainage system is usually characterized by the shape of its basin, stream order (e.g. *Horton* and *Strahler*) and drainage patterns of hydrography networks. For typical drainage patterns and descriptions of their possible generating processes, the readers are referred to [126, 159]. Some examples are shown in Fig. 2.2.

Within the domain of automated generalization, representing river networks is also an important topic. Efforts have been on preserving the structure (e.g. stream order), density, and shape (e.g. good continuity) while removing unnecessary details at a reduced scale [see e.g. 9, 197, 233]. Detailed characteristics of drainage patterns described above are considered during manual generalization carried out by cartographers [e.g. 184, p. 94],

but they have less been considered in current researches on automated generalization of hydrography.

Transportation networks

Structures in transportation networks evolve from basic connection patterns. Examples of connection patterns include 'ring', 'web', 'star', and 'hub-and-spoke' [228]. Note that these patterns only make use of topological information of a network, a ring or circuit network, for instance, does not necessarily mean a ring road, orbital motorway, beltway, circumferential highway, or loop highway in real world. The identification of such real-world phenomena (higher-level concepts) should further integrate metric information such as geometry, distance, orientation, and semantics.

Other useful reference to transportation networks can be found in [48, 115, 124]. In automated generalization domain, Heinzle et al. [91, 92, 93], Yang et al. [229], Zhang [232] have described the modeling and recognition of these structures from a technical standpoint.

Note however that, the requirements for groups such as networks are very difficult to formally define — only generic constraints on topological properties like self-intersection and connectivity can be defined for linear features [see 192]. It is still doubtable that whether such requirements are consolidated in human knowledge domain, not to mention the possibility to formalize them for automated processes.

Settlement structures and building patterns

Various kinds of knowledge such as historical, social, cultural and functional implications of built environments can be implied from characteristics, relationships and patterns of settlements and their structures embedded in topographic data/maps. This is because spatial configurations of settlements are the result of the same functional and social processes [48, 109, 115]. Examples of geographic processes that generate settlement are 'ribbon development' [119], urban development process, or urban morphology, such as 'perimeter block development' and 'street pattern' [112]. Besides, local building patterns can be used to infer other settlement information such as urban, rural, town center, commercial/industrial, and residential areas [184, 186] and land use types such as resort and military areas [75].

In this thesis, we will focus on building patterns in clusters (Sections 5.3

and 5.4) and on the evaluation of the generalization of building alignments (Section 6.4).

2.1.3 *Geospatial patterns and generalization*

Müller [135] noted that although it is intellectually interesting to build a scaleless, most detailed spatial database that allows users to zoom in at real-time to resolve the spatial conflicts by the overwhelm amount of information represented, this approach does not allow for the recognition of pattern generating processes at smaller scales. This pattern confirmation function of generalization was also noticed by MacEachren and Ganter [116], who suggested that a properly designed generalization should reveal rather than conceal important patterns. A poorly designed generalization would however give rise to misleading information.

According to Mackaness and Edwards [119], there are huge amount of patterns and higher-level concepts that can hardly be stored in spatial databases. However, geospatial patterns are only visible at certain scale intervals and for certain purposes. For example at larger scales settlement structures and building patterns are easier to identify, while network patterns become visible only when the scale becomes smaller. As a result, for a certain purpose and scale range, it is sufficient to focus on a limited set of patterns and higher-level concepts, which is more realistic for practical applications. This thesis work mainly focuses on structures and patterns of building features from large to middle scale (e.g. from 1:10k to 1:100k) and on constraints related to these structures.

2.2 Evaluation of generalization quality

This section gives a historical review of the state-of-the-art of quality evaluation in map generalization. Section 2.2.1 gives a general definition of quality evaluation in map generalization. Section 2.2.2 describes three different purposes of the evaluation. It further analyzes the benefits of these different purposes on different user groups (e.g. cartographers, data users, etc.). The following sections explain different early approaches that are found in literature to address the problem of quality evaluation in map generalization. These approaches include visual assessment (i.e. expert evaluation) in map generalization (Section 2.2.3) and some quantitative techniques (Section 2.2.4). Section 2.2.5 reviews the evaluation that is integrated as part of the generalization process. Section 2.2.6 presents the previous work related to the evaluation of generalization output.

2.2.1 *Definition of quality evaluation*

It is unavoidable that map generalization will change spatial data in term of data quality [135], but these changes could be helpful given a specific map purpose and scale (a fitness for use concept). Quality evaluation in map generalization is the process of examining and checking whether some 'desired' characteristics of a system or data are presented satisfactory for a given task. It is different from measuring the absolute loss in data quality which is not a problem in map generalization. Evaluation also provides users with necessary information to understand the ability and limitations of a system or data in use. It is acknowledged that evaluation is one indispensable component of a holistic generalization process in both manual or (semi-)automated generalization [72, 98, 117, 167, 217, 218, 220].

2.2.2 *Purposes of evaluation in generalization*

An earlier research on quality evaluation in map generalization was proposed by Weibel [218], who suggested that evaluation can be made at three different stages of the generalization process: a priori, posteriori and *ad-hoc* evaluation. Likewise, Mackaness and Ruas [120] summarized three scenarios where the evaluation may be applied. They are evaluation before, during and after generalization which are motivated by different purposes:

- **Evaluation for tuning:** it acts as a support for setting parameters and tuning generalization system before the processing.
- **Evaluation for controlling:** it acts as an intermediate step of the generalization process that occurs during the generalization. It can be used for (i) identifying conflicts and errors and (ii) improving the process by triggering the algorithm or modifying the sequence of operations applied.
- **Evaluation for assessing:** evaluation of generalization output after the generalization process.

According to Bard and Ruas [21], the last purpose, evaluation for assessing, can be further divided into three types:

- *Evaluate for editing* which aims to detecting cartographic errors and inconsistencies. It is performed at the final stage of the generalization process.
- *Evaluate for grading* which attempts to find an aggregated value that reflects the overall quality of the generalized data and makes comparison of different generalization solutions to determine the optimal one

or to identify poor generalization solutions given specific generalization tasks.

– *Descriptive evaluation* which provides general information of the modifications introduced to the generalized data. Such information can be used to improve the quality description (e.g. meta-data) of the final products (e.g. what has been removed, emphasized? How much is the data changed?)

The results of the quality evaluation are of benefit to different user groups. For the cartographers, the evaluation helps them detect errors during generalization process and facilitate the interaction between human and computer. For the system designers, the evaluation returns useful feedbacks which improve the development of automated generalization. For the data users, the evaluation results provide them with more knowledge on the content of the data in use.

2.2.3 *Visual assessment*

Traditionally, cartographers visually evaluated the quality of generalization based on map requirements. The judgement on the quality depends largely on their own knowledge and experience of cartography. The approach persists even with the (semi-)automated generalization. That is, the evaluation of resulting maps from automated solutions still resorts to visually comparing with, e.g., an 'optimal' solution (paper map), in order to validate the automatic solutions or algorithms and to identify cartographic conflicts.

Usually, visual (manual) assessment is carried out by asking the cartographic experts to grade the quality of solution based on certain criteria [120, 218]. The expert judgement can be made on a global criterion like 'maintenance of overall character of the original map', or the criteria can be a very detailed checklist. The grade varies from, for example, 1 to 5 representing the quality of generalization from 'poor' to 'excellent'. The quality of generalized map is determined based on the qualitative grading by different cartographers. The process is also called expert evaluation. The difficulty of expert evaluation is that the criteria defined for the generalization is too general and that different experts may have different opinions as to what makes a good generalization. Thus, it is much needed to set up a standard for the assessment. One approach to producing a comprehensive and common form of quality checklist and questionnaire is through a close collaboration with different experts [218].

The visual assessment is the simplest assessment to implement, but the

most expensive one if we are looking for a comprehensive assessment of the quality of a generalization [20]. Furthermore, visual assessment is subjective and qualitative and comments obtained from experts might be very general or based on fuzzy criteria [38, 98, 101, 120, 164, 221].

2.2.4 *Quantitative evaluation techniques*

In addition to the visual assessment approach, many researchers were seeking quantitative methods for evaluating the quality of generalization. The idea is to identify one or more quality measurements to determine the fulfilment of the product with respect to map requirements. Some early studies concentrated on the quantification of content change, while others focused on the measuring of specific primitive geometries before and after generalization.

Radical Law is one of the earliest attempts to formalize the change of map content over scale transitions [201]. *Radical Law* formulates the number of map objects (symbols) that should be selected for different scale transitions. As a result, the law can be used as a criterion for controlling the number of selection. However, the law cannot address the question of where and which objects should be selected, and consequently it is not capable of controlling semantic and structural aspects of generalized data. There was once a trend to utilize information theory (entropy) to model the changing levels of map content that arise from the generalization process [e.g. 31, 32, 33, 110, 141]. On the other hand, using information theory to evaluate the map content could be problematic. Mackaness and Ruas [120] for instance argued that it is extremely difficult to formalize the change of map content caused by generalization in terms of information entropy, since information of maps varies largely depending on map purpose, application of map symbols and map reader's interpretation skills.

Earlier attempts on the evaluation of primitive objects focus on linear geometries such as road, river, relief, railway, etc. McMaster [128, 129] proposed measures to quantify the position change of vertices of lines before and after generalization, based on the concept of 'vector displacement'. The main focus of this assessment is on geometric precision. A later work by Buttenfield [47] expanded the description of linear features to five measures (length, width band, segmentation, error variance, competition) which makes up the 'structure signature' of the lines. The aim of [47] was to formulate the changes of the 'structure signature' over scale transitions. These two contributions had inspired other attempts for describing the characteristics of linear feature, including geometric, structural aspects [8, 151, 206, 213]. In

addition, some studies have focused on evaluating the change of different types of linear objects in terms of line length and 'vector displacement' [e.g. 18, 98]. Finally, several studies suggested a methodology for evaluating the quality of generalized lines based on the description of the structure and shape of the lines [e.g. 178, 179].

The work of evaluating polygonal objects is undertaken more recent than those of linear objects. Two types of objects are treated separately: buildings and land use. These two types of features are differentiated by map requirements due to their different properties in shape and topology. The building features are man-made constructs with regular shape and they are usually spatially disjointed objects; whereas land use features have irregular shape and are adjacent to each other. Whether these different properties are maintained or not have to be evaluated in different ways. Regnauld [155] focused on the generalization of buildings. He argued that there is a need to evaluate whether the generalization results correspond to the following three aspects: (i) characteristics of buildings (e.g. size, ratio between the number of buildings at the target and initial scale), (ii) spatial distribution of buildings (e.g. alignments and patterns), and (iii) density of buildings. Other measures for evaluation of polygonal generalization can be found in Peter [148]. Dettori and Puppo [57] proposed to extend the quality evaluation of polygonal objects with topological requirements (e.g. connectivity, inclusion, etc.).

However, evaluation of primitive geometries is not enough, since multiple objects are involved in the quality of generalized map [220]. As a result, relationships between objects (either within a thematic class or between different classes) should be formalized and taken into account by the evaluation process.

2.2.5 *Evaluation for controlling the generalization process*

Earlier theoretical analysis on evaluation for controlling a holistic generalization process was made by McMaster and Shea [132], who suggested using 'cartometric evaluation' to answer the question of when to generalize. They further suggested to use a series of 'spatial and holistic measures' (e.g. measurement of density, distribution, shape, etc.) to detect undesired situations (conflicts) and select operations to resolve those identified conflicts. Later, Ruas and Plazanet [167] and Weibel and Dutton [220] proposed similar generalization models which treat automated evaluation as an integral part of the generalization process, based on the idea of cartographic constraint (i.e. map specifications, see also Section 2.3.2). In their proposed

models, the evaluation is used for structure recognition, conflict detection and quality assessment [1]. Brazile [38] designed an architecture of automated generalization incorporating quality assessment for controlling the whole process. At the moment, there is no implementation for those proposed conceptual models due to the complexity of the process and the limitation in technologies.

With the development of constraint-based approach, more and more advanced techniques are adapted from other domains to automated generalization. Multi-Agent System (MAS) for example is a technique to implement constraint-based generalization, in which evaluation can be performed before and after each step of the generalization in order to reach the optimal solution for designed constraints. The research of e.g. Galanda [72], Galanda and Weibel [73] focused on the generalization of categorical (land use) data with MAS technique. The evaluation mainly focuses on the calculation of some basic metric and topological constraints, such as keeping minimum distance and avoiding self-intersection. Based on measured values and their discrepancy to the goal values, the violation (or satisfaction) of a constraint can be calculated and a list of generalization plans is proposed automatically. On the other hand, optimization techniques were also widely used to implement constraint-based generalization [e.g. 84, 86, 89, 172].

In summary, quality evaluation is an indispensable component for building a successful automated generalization system. The evaluation as a part of a generalization system has the advantage of informing the system whether it reaches the final goal (optimal solution) directly. But it has two disadvantages: on the other hand, evaluation criteria (i.e. constraints) might be the same as the constraints used to steer the generalization process. In such a case, generalized data usually satisfy the evaluation criteria. The evaluation as such may give over-optimistic results [88], because the data may violate other important constraints that are not defined. On the other hand, the comparison between different generalization outputs (i.e. Evaluation for grading) performed on different systems is not possible for systems with self-evaluation capabilities [164].

2.2.6 *Evaluation for assessing generalization output*

As described in Section 2.2.5, evaluation for controlling is not a good option for assessing the overall quality as well as making comparison between different solutions. To achieve the goal of assessing the final output of generalization, an independent evaluation is necessary. Only few studies that related directly to the evaluation of generalized data at the end of the

process was carried out by previous researchers.

The first contribution to this topic is that of Ehrliholzer [65]. The important ideas emerge from this previous study are: (i) the use of three levels of abstraction for assessment (i.e. element, group and map level), (ii) the need to standardize the scale of various criteria (e.g. a value from 1 to 5) for comparison purpose, and (iii) the combination of criteria to provide a summary evaluation results. The study remains a theoretical one and only some generalization algorithms were evaluated. Other experiments such as the ones developed in the OEEPE project [164] and AGENT [2] project helped the community to develop various evaluation techniques.

Up to date, the most comprehensive study on the automated evaluation of generalized data was undertaken by Bard [19, 20]. One of the objectives of his study was to develop a synoptic evaluation framework in which the relative quality of generalization outputs can be graded (Evaluation for grading). The author identified a set of reference functions for automated evaluation and the interaction between object types, object characters, reference functions and threshold values of these functions according to the scale transition. The author evaluated map requirements on buildings to show how these functions work in evaluation process. A more recent research on the evaluation of generalization output was carried out in the EuroSDR project [190], see Section 2.4 for a short description.

2.3 Automated evaluation of generalization output

Visual evaluation is subjective, qualitative, and time-consuming. The quantitative assessments described previously only examine certain characteristics and are therefore valid only for assessing specific requirements. Also, those techniques can hardly be adapted and integrated to a holistic automated generalization process. Comprehensive solution for automated evaluation would be of great importance. The research on automated evaluation in map generalization is still at its starting point [45].

This section explores the topic — automated evaluation of generalization output after the generalization process, which belongs to the third category of the purposes of evaluation, i.e., Evaluation for assessing (Section 2.2.2). This topic is the main focus of this thesis work. This section aims to provide an overview of what have been done for automated evaluation of generalization output and what are left and hence need to be done. Section 2.3.1 gives an overview of the major components of automated evaluation of

generalized data. The following sections further explain these components, namely defining and formalizing the map requirements (Section 2.3.2), measures for automated evaluation (Section 2.3.3) and the need for data matching to enable the automated evaluation and current development of data matching techniques (Section 2.3.4). These sections outline the major difficulties of this topic.

2.3.1 *Previous framework and components*

As suggested by Weibel [218], the overall process of evaluation involves two tasks (or components):

- Formalization of map specifications: map specifications must be formalized to describe what is a 'good', 'acceptable', or 'bad' generalization for a given application.
- Evaluation process: applying methodology to determine the degree to which the specifications are respected by a given generalization output.

The above two components were mentioned also in the work of e.g. Bard and Ruas [21]. Mackaness and Ruas [120] further pointed out that a challenge of automated evaluation is to decompose the evaluation process into a set of values that can be calculated automatically and that remain meaningful to users. Therefore, it is important to know which characteristics are pertinent to a specific map task and scale and how to define them precisely (formalization). Then, it is also important to develop various techniques to qualify and quantify the actual changes of characteristics of data objects, which are later compared with the formalized map requirements (evaluation process). In AGENT project [1], evaluation process was further divided into three steps: (1) measurement, which is used for the computation of raw parameters (e.g. area of buildings), (2) characterization, which aims to interpret those measured parameters (e.g. using Min, Mean, and Max values of computed areas to classify buildings), and (3) evaluation, which determines the characterization by comparing the measured value and goal value (e.g. if a building is too big or too small with respect to the specifications).

Bard and Ruas [21] proposed a comprehensive framework of automated evaluation of generalized data which extends the work of AGENT [1]. The components of their framework and the interactions among the components are shown in Fig. 2.3.

In this evaluation model, **Specification** (map requirements) contains lists

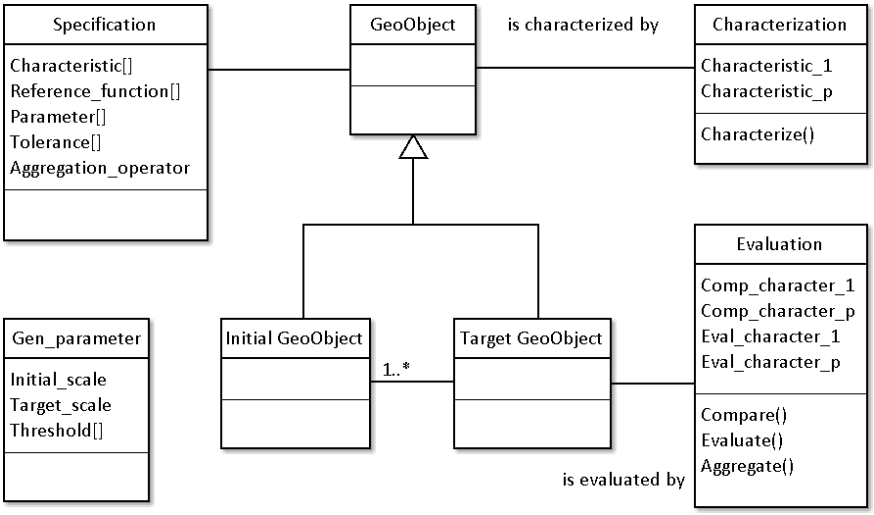


Figure 2.3 The model to evaluate generalization output (by Bard and Ruas [21])

of characteristics, reference functions, parameters, etc. The establishment of map specifications is referred to as the formalization process mentioned above. In Bard and Ruas's work, the specifications mainly contains formal requirements of buildings (e.g. size, shape, orientation, etc.). Their focus is on the framework development and the evaluation process. In their model, the evaluation process is modeled as two classes, namely the characterization and evaluation. **Characterization** holds the measures (algorithms) to calculate the properties of geographic objects (both initial and target objects), such as size, shape and orientation of buildings. **Evaluation** first uses `Compare()` method to calculate the distance between measured value of a characteristic and the target value defined in the specifications, then it revokes `Evaluate()` to interpret the distance into a qualitative evaluation result according to the defined tolerance. For example, if the distance between measure size and target size of a building is 15 map mm^2, and the tolerance is 20 map mm^2, then the generalization of this particular building is 'good' in terms of size property.

It is noticeable that the evaluation process in this model involves measuring and comparing properties of objects in both initial and target scales. That is to say, corresponding relations between target and initial objects should be explicitly modeled. Burghardt et al. [44] and Stoter et al. [192] also suggested that the evaluation of generalization output should be based on the specification defined as a set of constraints, and partly on the initial

dataset. As a consequence, data matching for automated evaluation of generalization output is needed.

2.3.2 *Defining and formalizing map requirements*

In general terms one can define requirements for generalization outputs as, e.g., reducing the details to discern regional patterns; an aesthetically pleasant map; a map that reveals or conceals information inherent among a set of abstracted data [120]. The difficulty is that these map requirements should be specified in such a way that they can be interpreted by computer if one wants to evaluate against these requirements automatically [44, 192]. Stoter et al. [192, 193] further pointed out that some map requirements are only available at human knowledge level. In this context, automated evaluation faces a bottleneck of knowledge acquisition[2] and formalization.

This section first reviews the impediments for the formalization process in terms of the various heterogeneous knowledge sources used (Section on 'KNOWLEDGE SOURCES FOR MAP SPECIFICATIONS'). Then previous efforts on capturing various sorts of generalization knowledge is introduced in the Section on 'KNOWLEDGE ACQUISITION TECHNIQUES FOR MAP REQUIREMENTS'. This section also gives an overview of knowledge engineering techniques adopted in automated generalization domain, which are promising for defining and formalizing map requirements for automated evaluation. Section on 'SPECIFYING MAP REQUIREMENTS IN CONSTRAINTS' describes current development of using constraints to specify map requirements for automated generalization and evaluation. Section on 'TYPOLOGIES OF CARTOGRAPHIC CONSTRAINTS' describes previous attempts to categorize map requirements in the form of cartographic constraints. Section on 'DIFFICULTIES IN FORMALIZING MAP REQUIREMENTS' reports the difficulties in the process of formalizing map requirements.

Knowledge sources for map requirements

To define and formalize map requirements, a complete view of generalization knowledge is needed. A full spectrum of the knowledge of map requirements are covered by various sources such as textbooks, NMA specifications, spatial data models, program code, expert knowledge [44, 191], as well as map series, cartographic literature or documents in projects such as AGENT and EuroSDR.

[2] Knowledge acquisition denotes the process of encoding problem-solving expertise from a knowledge source to a formal structure readable by software [127].

However, not all of the map requirements are presented in such a formal and specific way that enables automation. Some of them are only available at human knowledge level, and thus the evaluation of such requirements depends largely on human interpretation [191]. For example, the study of cartographic literature can provide a starting point for establishing the map requirements for evaluating different generalization alternatives. Many textbooks (e.g. SSC [184]) and NMA specifications employ 'good' and 'poor' generalization examples to illustrate the map requirements. These examples only describe certain aspects of the generalization on the one hand. On the other hand, they are intuitive and graphic-based [218], which means that most of them are not computer-readable.

Given the difficulties in utilizing the knowledge sources described above for defining map requirements, one has to use all available techniques to extract useful knowledge from these sources, since they are the only sources for defining and formalizing map requirements to enable automated evaluation of generalized data.

Knowledge acquisition techniques for map requirements

Knowledge acquisition techniques were previously widely studied to develop knowledge-base system (KBS) for automated generalization. The bottleneck of KBS is the decomposition of the generalization knowledge into a set of applicable rules that are capable of steering automated generalization process [222]. The knowledge of generalization can be divided into three basic types, namely geometric, structural and procedural knowledge [16, 222]. Geometric knowledge is used to describe geometric aspects of individual objects (e.g. size is too small to be legible). Structural knowledge describes properties of groups of objects that are perceived as an organization (e.g. building clusters and alignments). The procedural knowledge is about which generalization operations to apply in different situations, which is usually rule-based. Different approaches were adopted to capture these types of generalization knowledge from the sources mentioned in the above section:

- **Expert interview** is an attempt to extract knowledge in experts' mind, which can be achieved by different methods such as structured interviews, repertory grids, critical incidents or artificial problems [102, 127, 145].

- **Reverse engineering** aims to collect knowledge from map series by comparing map objects over scale transitions [47, 107].

- **Interactive system** (i.e. amplified intelligence) aims to analyze the interactive generalization process and learn from the behavior of cartographers [130, 157, 217, 222].
- **Machine learning** attempts to derive rules by learning from a set of examples (training data) provided by experts or by self-learning (clustering) without training data [138, 151].

To drive the power of rule-based generalization system, procedural knowledge needs to be formalized into rules to mimic how cartographers generalize maps. Hence, more attentions have been paid to the acquisition of procedural knowledge. The rules are usually formalized in the following form: `if condition [object area` $\leq$ `1000], then action [enlarge object].` The investigation of the above-mentioned techniques, especially the amplified intelligence technique, is initiated for the development of rule-based system. Despite few rule-base generalization prototypes established for specific well defined problems such as label placement, there was no proof that a rule-based (or knowledge-based) generalization system can be built [137].

Specifying map requirements in constraints

Beard [23] pointed out several drawbacks of rule-based approach to automated generalization and for the first time introduced a constraint-based approach, which makes the automated generalization more flexible and promising [22, 167, 220]. For instance, a constraint requires that minimum distance between two buildings should no less than 0.2 map mm, and different operators can be applied to achieve the goal in case of a violation. More importantly, cartographic constraints become a natural way to explicitly describe the objectives (quality criteria) of generalization output. Hence, constraints on a feature class, a situation (a group of objects), or an individual object allows the quality of each level to be assessed [44, 72, 185, 191].

From the perspective of knowledge engineering, constraint-based approach avoids the difficulty of acquiring and formalizing the procedural knowledge (if-then-else rules). However, the acquisition of other types of knowledge (e.g. geometric and structural knowledge) remains an open issue for constraint-based approach. For example, the shape constraint requires the knowledge on how to mathematically define shape for the purpose of generalization. Such knowledge may be in the mind of experts or contained implicitly in map series.

The acquisition and formalization of constraints are not only important

for automated constraint-based generalization but also for automated evaluation of generalization output. As remarked by Mackaness and Ruas [120], a method is needed for calibrating and prioritizing map requirements for the purpose of automated evaluation. Thus, the constraints for automated evaluation have to be well defined. Knowledge acquisition techniques such as expert interview, reverse engineering, and machine learning provide potential solutions for extracting and formalizing map requirements from all available sources.

It is important to distinguish between cartographic constraint and the constraint concept in database domain. An integrity constraint in database domain is a condition that has to be met strictly for all data objects. Cartographic constraints on the other hand can be further divided into soft and hard constraints. A soft constraint is a goal that can be optimized. For example, the goal to ensure the minimum sizes of buildings for a better visualization can be relaxed under over-constrained situations (e.g. settlement population is too dense), or to meet other more important constraints (e.g. keep important buildings). A hard cartographic constraint (e.g. data should be topologically correct) is similar to the constraint in database domain, that is, it has to be satisfied strictly. To balance between different cartographic constraints, importance values should be assigned to weight different constraints for automated evaluation.

Typologies of cartographic constraints

Previous typologies of map requirements are based on the notion of cartographic constraint, as constraints define explicitly the requirements of generalization output without addressing how the result should be achieved. Beard [23] proposed the first classification of constraints by categorizing them into graphic, structural, application and procedural aspects. Ruas and Plazanet [167] refined Beard's typology and suggested that there are geometric (e.g. legibility and shape), spatial relationship, and semantic relationship constraints. A comprehensive categorization is shared by some authors and projects. The categorization comprises of five major categories: graphic (legibility), topological, structural, Gestalt, and procedural constraints, where structural constraints describe higher-level concepts and Gestalt constraints deal with the aesthetical aspect [1, 72, 220].

In favor of automated evaluation of generalization output, EuroSDR-project (see Section 2.4) proposed a typology of constraints which extends the previous typologies [see 44]. The typology distinguishes between two major types of constraint, namely the legibility (readability) and the preservation

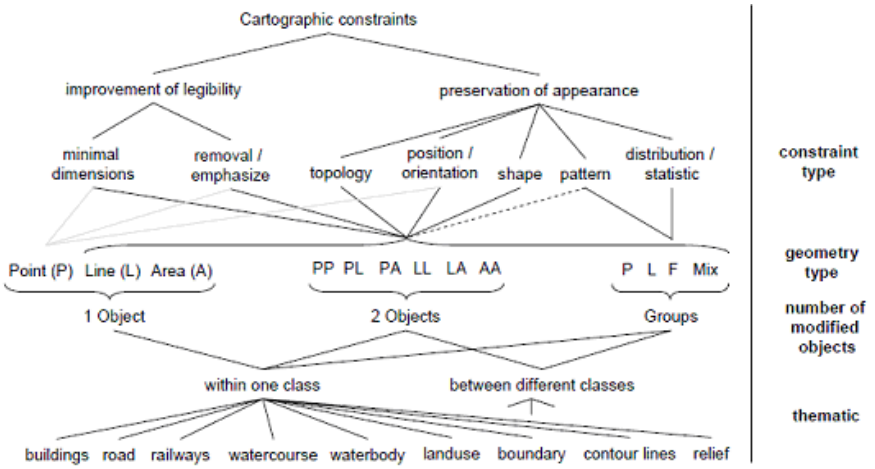


Figure 2.4 Typology of cartographic constraints (from Burghardt et al. [44])

of appearance constraints. The legibility constraints can be evaluated solely based on specifications, while assessing the preservation constraints has to measure properties on both initial and target objects. In addition, the typology takes into account the constraints on different number of objects. Three categories are identified: constraints on one object, between two objects and on groups of objects. Constraints on one object contain minimal size, self-intersection, absolute position, orientation, shape, etc. Constraints between two objects include for example minimal distance, topology, and relative position (e.g. orientation relationship among more than two objects). Constraints on pattern (e.g. alignment), spatial/semantic distribution, density and network features apply to groups of objects or a whole feature class [21]. The constraints on groups of objects are defined at the meso- or macro-level and often deal with higher-level concepts, and they are harder to formalize than the constraints on one object [44, 192, 219]. The hierarchy of constraint type, geometry type, number of modified objects and thematic type is shown in Fig. 2.4.

Difficulties in formalizing map requirements

In automated evaluation of generalization output, the aim of formalizing map requirements is to know which constraints are appropriate to define the quality of generalized data and how the constrained properties change at scale transitions. The former aims to select a set of constraints, whereas the latter aims to specify the goal values for the constraints (e.g. the min-

imum area of buildings after generalization). Specifying goal values for constraints is however not always straightforward, since it depends on the one hand on the specifications of generalization, and on the other, it based partly on the initial state of the property [44]. Results from the EuroSDR project [192] shows that the number of constraints defined on two objects and on groups of objects are more difficult to interpret and evaluate by commercial generalization systems than those defined on individual objects. This observation reveals the current situation of the formalization of map requirements. The situation can be improved by paying more attention on the formalization of such constraints.

A closer look at the literature shows that, low-level constraints such as those on one object (e.g. minimal dimensions) and between two objects (e.g. topology) are relatively easier to formalize. However, the constraint on general shape, which is also a low-level constraint, is hard to formalize. Shape is the ability to recognize the original shape of objects and is different from position accuracy [137]. So shape needs to be preserved during generalization as long as the details are still readable. In the constraint 'target shape should be similar to initial shape', shape is not a mathematically defined concept and the operation 'similar' is not well-defined either. Defining and precisely measuring shape is necessary for automated evaluation. The 'similar' of two shapes can be formalized by specifying the scale-dependent modification to the measured values. A potential way of acquiring the scale-dependent knowledge is by studying map series at different scales (i.e. reverse engineering) as indicated in Harrie [84]. The results of the acquired change of a property can be stored in an evolution function as proposed in [19].

In addition to the shape constraint, the constraints dealing with higher-level concepts such as pattern, distribution, and network are also difficult to formalize. Most of these constraints act as restrictions to the generalization of groups of objects. These higher-level map requirements are important for defining the quality of generalization output over large change in scale, since the concepts describe important characteristics of the map at smaller scales and hence should be visually more discernable at the scales [119, 120, 135, 166]. At the conceptual level, Steiniger [185] proposed to formalize patterns using horizontal relationships, which describe the interactions between map objects within the same map scale. The aim of the formalization of patterns in his research is to recognize patterns, which is a first step of the formalization of patterns for automated evaluation. The recognition of patterns in this thesis is viewed as a data enrichment

process. A complete view of the formalization of higher-level constraints for automated evaluation should also involve modeling the change of these concepts at scale transitions. This means that how the concepts modify themselves as scale transitions should also be formalized. However, the knowledge of these constraints is previously not expressed in a clear and computer-readable way. Therefore, knowledge acquisition techniques are still needed for extracting from various sources the knowledge for automated evaluation of generalization output.

2.3.3 *Measures for automated evaluation*

As described in Section 2.3.1, automated evaluation can be described separately by three major tasks: the first one is defining and formalizing map requirements; the second one is the measures which automatically quantify the properties that need to be evaluated; the third one is evaluating by comparing between corresponding objects or with respect to the specifications. Measures play an important role in automated evaluation: they are used to detect the constraint violations and describe the severity of a violation. For example, a target building with an area slightly larger the size threshold is more acceptable than if it is smaller than the threshold. Using sensitivity ranges (tolerances), a measure can evaluate the severity (degree) of a violation and the violation can be easily interpreted by human subjects.

Section on 'DEFINITION OF MEASURE' gives a definition of measure that is useful in the generalization domain. Section on 'CLASSIFICATION OF MEASURES' reviews previous classifications of measures that are relevant to quality evaluation. Section on 'MEASURES OF SHAPE' and 'MEASURES OF HIGHER-LEVEL CONCEPTS' present respectively the previous efforts on measuring shape and higher-level concepts.

Definition of measure

A measure is defined in AGENT [2] project as a procedure to compute measurement, which is the basis for evaluating characteristics of map objects and assessing the need for generalization (conflicts detection) and the success of generalization (evaluation of generalization output). Measures can be understood at four aspects. Take distance measure for example, it has an operational aspect (distance), a mathematical aspect (*Hausdorff* distance), an algorithm (implementation of *Hausdorff* distance) and a measured value. A measure can be used to quantify a property of an object with respect to a constraint. A measure may consist of a mere formula or it involves a complex algorithm including computation of representations

and auxiliary data structures. Representations here describe the geometry of object(s) derived by a transformation from 2D Euclidean space to another reference-system (e.g. Fourier transform) or by a transformation between different dimensions (collapse operation). Auxiliary data structures capture the relationships among a group of objects within higher-level concepts such as alignments and networks. Examples of these data structures are Delaunay triangulation and Minimum Spanning Tree (MST) [2].

Note that mathematical measures can be exact or approximate, according to the implementation algorithm. To begin with, we can use either approximate *Hausdorff* algorithm or precise ones to calculate the distance between two lines. But computing precise *Hausdorff* distance is time-consuming. Second, the values or measurements of measures have different types: numeric (e.g. distance(a, b) = 3.2) and Boolean (e.g. disjoint(a, b) = False). Third, it may not be adequate to fully describe higher-level concepts like patterns and structures using a single value. In many cases they have to be characterized geometrically using auxiliary data structures such as MST [e.g. 154].

Classification of measures

McMaster and Shea [132] described several measures in digital cartography for determining when to apply generalization operations: measure of density, distribution, length and sinuosity, shape, distance, Gestalt, and abstraction. To support quality evaluation, Weibel [218] proposed a classification of measures: global, geometric, topological, and software-related measures. Up to date, the most comprehensive classification of measures for quality evaluation is perhaps the one proposed by Mackaness and Ruas [120]. It has two aspects:

- A measure can be internal or external:
  - Internal: measure for object(s) at the same scale (within a dataset)
  - External: measure for objects between two scales (before and after generalization)
- A measure can be Micro, Meso or Macro:
  - Micro: measure on individual objects or part of the objects
  - Meso: measure on groups of objects
  - Macro: measure on all objects of a feature class

As defined in AGENT [2] project, there are actually two classes of external measures. On the one hand, implicit external measure is computed by comparing two internal measurements calculated at both scale independently (e.g. change rate of size by dividing the two areas). On the other hand, explicit external measures really compare two objects at different states (e.g. *Hausdorff* distance for describing the shape change of lines at two scales). The measures at micro- and meso-level are in line with the map requirements on one object, between two objects or on groups of objects. For instance, measures at micro-level relate to position, orientation, size, and shape constraints on one object; measures at meso-level deals with topology, spatial arrangement/distribution, density constraints on groups of objects.

Measures of shape

Shape describes the geometric form of individual spatial objects [223] and should be maintained during the generalization process. General shape is difficult to measure. Lots of internal shape measures (e.g. sinuosity, complexity, elongation, compactness, concavity, shape index, etc.) have been summarized in AGENT [2]. The measures describe certain aspects of shape by assigning these aspects with scalar values. There are also similarity measures of shapes that have been studied. Veltkamp and Hagedoorn [211] summarized several techniques for shape matching (e.g. *Fréchet* distance, *Hausdorff* distance, turning function, signature function, etc.). Ai et al. [10] proposed a shape descriptor for polygonal objects based on Fourier transform. These measures can be used for comparing the similarity of shapes at different map scales.

To use the above-mentioned various measures, some questions arise naturally: Which is the best suited measure for the description of general shape in map generalization? If there is no such a best measure exists, then whether and how can we combine different measures together to describe general shape approximately? If such questions are answered, shape measure is meaningful to automated evaluation of shape constraint [45].

Measures of higher-level concepts

What is high-level concept in the context of cartographic generalization? To define it in a strict way seems to be not possible. We enumerate several typical higher-level concepts (patterns) in the geospatial domain in Secion 2.1, and discuss their relationship to map generalization. High-level concepts or phenomena are especially important at smaller scales. The next ques-

tion is what are the characteristics of these high-level concepts? Can these characteristics be measured? Do these measures really reflect the nature of the concepts in experts' mind? How do these high-level phenomena change at scale transitions? These questions are the most pertinent to automated evaluation of generalized data in terms of higher-level concepts. As stated previously, measuring higher-level concepts might involve complex computations of auxiliary data structures, such as Delaunay triangulation, MST, Voronoi diagram, etc. These data structures are used to support the measures on higher-level concepts.

Different aspects of the measure on higher-level concepts have been studied. Regnauld [155] used MST in conjunction with road network to measure the characteristics of groups of buildings. Christophe and Ruas [53] used a straight-line scanning technique and statistical analysis to detect building alignments and measure their distributions. For network features, 'continuity' is regarded as a measurement for evaluating the quality of generalized network feature [61]. Other methods for describing network feature basically utilizing graph theory [91, 92, 93, 97, 118].

Given current progress in measuring and modeling the higher-level concepts however, it is not known yet which one is suited best for a certain characteristic with respect to a constraint. In addition, little work is found in formalizing the modification of these high-level concepts at scale transitions. This is a research challenge for this thesis work.

2.3.4 *Data matching*

One difficulty of automated evaluation is that the reference data and the data being evaluated are at different scales, and the reference data are over detailed [21, 120]. Consequently, corresponding relationships identifying objects that represent the same real-world entities (an object or a group of objects) at different scales [34]. In order to measure the properties of spatial entities at different scales and then compare the measured values of both entities, the relationships between entities at two scales should be explicitly modeled and stored in databases.

In the generalization domain, this kind of relationship is also named vertical relation which is a kind of data enrichment [142]. The vertical relationship has three major different classes: 1-to-1 relation, n-to-1 relation and n-to-m relation [4]. The identification of these different types of vertical relationships is of great importance to the evaluation of generalized data [21, 34, 120]. The 1-to-1 relationship is the simplest relationship that

can be identified during generalization. The automated evaluation against preservation constraints on one object (e.g. absolute position, orientation, shape) depends on the 1-to-1 relationship of corresponding objects. The 1-to-1 relationship is usually resulted from generalization operations such as simplification, exaggeration and displacement [143]. The n-to-1 and n-to-m relationships are usually resulted from applying generalization like aggregation and typification.

The vertical relationships of topographic data at different scales are not an explicit and standardized part of most data models. The links have to be created by means of automatic matching techniques. Nowadays, the mostly discussed objectives of data matching between different levels of detail are for the establishment of MRDB and spatial database integration [27]. Lots of studies on the data matching of network data (e.g. road network) can be found in the literature [58, 139, 231].

It is clear that automated evaluation against the constraints on 1-to-1 relationships (e.g. preservation of absolute position, orientation, etc.) will benefits from data matching process. But it is not clear yet to what extent the data matching is required for automated evaluation and how this works for n-to-1 and n-to-m relations. We will present our approaches to matching spatial entities (individuals and groups) for automated evaluation in Chapter 6.

2.4 EuroSDR project

EuroSDR project on the state-of-the-art of automated generalization is a cooperative project among Universities, NMAs, and institutes across Europe. The main objective of EuroSDR project is to study the state-of-the art of currently available commercial generalization software packages in their capabilities of generalizing a complete map. The map specifications are defined by the involved NMAs and harmonized before use [192]. Additionally, the project also aims to get insight into several side aspects, such as how to specify requirements for generalization, how do generalization processes work, how to set up a case to study the-state-of-the-art in generalization, how to perform evaluation of generalization output? [190, 192].

The methodology of the project consists of (1) setting up the test cases, (2) involving commercial generalization software vendors into the project, (3) defining map requirements, (4) setting up test process, i.e., test assignment between project members, and (5) evaluation methodology. In the

Area type	Source dataset	Target dataset	Provided by	Feature classes
Urban area	1:1250	1:25k	OS Great Britain	Buildings, roads, river, relief
Moutainous area	1:10k	1:50k	IGN France	village, river, land use
Rural area	1:10k	1:50k	Kadaster, NL	small town, land use, planar partition
Costal area	1:25k	1:50k	ICC Catalonia	village, land use (not mosaic), hydrography

Table 2.1 Test cases in EuroSDR project on automated generalization (from [192])

following, we shortly describe the test cases (datasets) used in this project (Section 2.4.1), cartographic constraints harmonized for automated generalization (Section 2.4.2), and evaluation methodology proposed in the project (Section 2.4.3). The project is one of the context of this PhD research and partly motivated this thesis. Furthermore, part of the datasets from the project are also used in this thesis to test the presented automated evaluation framework and its components.

2.4.1 *Test cases*

There are four source datasets (Table 2.1) and 3–4 generalization systems involved. The generalization test is carried out by up to 4 testers. Around 40 generalization outputs were generated from the test process.

2.4.2 *Harmonized map specifications*

In this project a comprehensive list of constraints was gathered based on an expert survey to largely capture map specifications of NMAs across Europe. Initially about 250 constraints were collected for the four test cases (Table 2.1) which usually contains overlapping situations. Then a structured template was developed to harmonize the constraints, which results in 45 generic constraints [192]. The structured template is able to record information items needed to define constraints for automated processes (see e.g. Fig. 2.5). Basic items such as constraint type (e.g. preservation), subject to be evaluated (e.g. feature class or kind of group), constrained property (e.g. size), conditions to be respected (e.g. thresholds) are contained for all constraints. Other information may be required depending on the type and the scope of the constraints (e.g. a constraint between two objects needs

(a)

Geometry type	Class	Conditions for object being concerned with this constraint	Constrained property	Condition to be respected

(b)

Geometry type combination	Class 1	Conditions for object in class 1 being concerned with this constraint	Class 2	Conditions for object in class 2 being concerned with this constraint	Conditions for both objects (in the initial data) being concerned with this constraint	Constrained property	Condition to be respected

(c)

Geometry type combination	Class	Kind of group	Kind of objects of the initial data composing the group	Conditions for group (in the initial data) being concerned with this constraint	Constrained property	Condition to be respected

Figure 2.5 Simplified version of the structured template of generic constraints in the EuroSDR project: constraint structure (a) for one object, (b) on two objects and (c) for group of objects

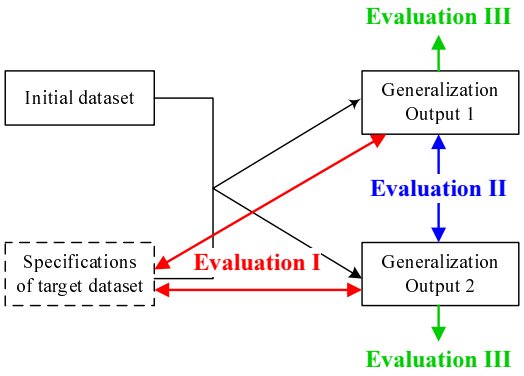


Figure 2.6 Overview of the evaluation methodology developed in the EuroSDR project (from [44])

the information of the second subject). However, not all the constraints were fully formalized with this template. For example, the condition 'target distribution should be similar to initial distribution' is not as formal as 'target size should larger than X mm^2'.

2.4.3 *Evaluation methodology*

Evaluation methodology in EuroSDR project is related to the evaluation of generalization output after the generalization process. It contains three basic evaluation procedures as shown in Fig. 2.6. The three types of evaluation are explained in the following.

- **Evaluation I**: automated evaluation of generalized data depends on the specification of target dataset (defined as a set of constraints), the results and partly on the initial source dataset. This type of evaluation is to answer: how many constraints are solved or violated? To what extent is the generalization output satisfactory with respect to the map specifications.
- **Evaluation II**: this type of evaluation compares the results of different outputs generalized by different software, in order to get insight into the generalization process.
- **Evaluation III**: this type of evaluation requires experts to visually evaluate the final results (expert evaluation).

The topic covered in this thesis belongs to the first type of evaluation. It is worth mentioning that within the context of EuroSDR, automated evaluation was carried out only for constraints on minimum size and minimum distance [45, 190, 192]. Due to the huge amount of work needed to evaluate all the generic constraints for all feature classes, geometry types, etc., it is

not possible to automatically evaluate all of the constraints within the time frame of the EuroSDR project [45]. Following up research was carried out to evaluate several preservation constraints on individual objects [see 42, 171].

2.5 SUMMARY

The automated evaluation framework described in this thesis targets at the purpose of evaluation for assessing. Of course the techniques presented are also beneficial to other types of evaluation.

In summary, three major components are identified for automated evaluation of generalization output. One is the definition and formalization of map requirements, and the second one is measures used for quantifying characteristics of map objects or phenomena. In this thesis, the information needed by automated evaluation contains not only characteristics and relationships to be measured, but also higher-level concepts, semantics, entities, and implicit geometries to be enriched (e.g. patterns to be recognized). As a result, we prefer to use a broader term, data enrichment, to describe this component. The last one is the data matching between corresponding objects. An improved framework for automated evaluation of generalization output is proposed in Chapter 3.

We can also conclude from the literature that, previous approaches to automated evaluation focused mainly on readability constraints that concentrate on one objects and between two objects. This is evident in the evaluation model by Bard and Ruas [21], where the model focuses on individual objects (Fig. 2.3). Entities such as groups and feature classes and characteristics of patterns and structures were previously less studied. This is mainly because previously constraints on groups, features, patterns, networks, and spatial distributions are not well defined [44, 192]. In addition, the evaluation of preservation constraint which need the maintenance of complex (n-to-1 and n-to-m) corresponding relationships between identical spatial entities was previously less implemented. In this thesis, we will address the evaluation of preservation constraint especially on groups and feature classes, where related patterns and structures will also be evaluated. A detailed formal model of our evaluation framework is presented in Chapter 4, which aims to bridge the above-mentioned gaps and provides a more comprehensive framework for automated evaluation of generalization output.

3 A conceptual framework for automated evaluation

Berners-Lee [26] made his argument that 'A Semantic Web is not Artificial Intelligence' (see also [108, p. 2]). Similarly, we realized that automation or machine-based processing is not equal to intelligence neither, that it means the use of well-defined rules (specifications) and well-defined operations on existing data to solve well-defined problems, and that it requires people to make the extra effort. If a problem requires information that is not explicitly available in existing data, the automation cannot be realized immediately. Similarly, if the specifications and operations to solve a problem are not well-defined, the automation cannot be realized as well.

The same is true for automated evaluation of generalization output. First, the map requirements (specifications) that define the generalization are not always well-defined. Second, the information contained in general-purpose topographic data sets are locations (geometries/corrdinates), feature types, and basic attributes such as length and size. However, extra information that is not available in existing data is still needed for automated processes. The extra information includes properties of spatial objects that are not stored in the database and implicit characteristics of or relationships between objects and patterns of object groups. In general, the extra effort has to be made to enrich the existing data either by adding various types of information off-line and storing them for later use, or by calculating them on-the-fly.

This chapter presents a conceptual framework for automated evaluation of generalization output. Section 3.1 gives an overview of the conceptual framework, after which some of the key concepts in the framework are further explained. Map specifications are a basis component for the framework and they defines the need for data enrichment and matching and the goal of the evaluation. As a result, we firstly analyze aspects of map specifica-

tions which gives insight into the formal structure of map specifications (Section 3.2). Section 3.3 then describes how map specifications (constraints) are evaluated and converted into quality descriptions. In Section 3.4, we summarize guidelines for a set of evenly balanced constraints to avoid over-optimistic evaluation results as discussed in Section 1.2.2. Finally, we refine the proposed conceptual framework and propose a three-step approach to automated evaluation of generalized data in Section 3.5.

OWN PUBLICATIONS

This chapter is partly based on the following publications:

- X. Zhang, J. Stoter, and T. Ai. Formalization and automatic interpretation of map requirements. In *The 11th ICA Workshop on Generalisation and Multiple Representation*, page 12, Montpellier, France, 2008.

- X. Zhang, J. Stoter, T. Ai, M.-J. Kraak, and M. Molenaar. Automated evaluation of building alignments in generalised data. *Manuscript submitted for review to an academic journal*.

3.1 FRAMEWORK OVERVIEW

The proposed framework of evaluation is outlined in Fig. 3.1. There are five major components in this conceptual framework. The existing data component are topographic data including initial and target (generalized) data. Target datasets are the subject of the quality evaluation. The two intermediate components are data enrichment and data matching units. The two units are used to make sure that the information needed by the evaluation component is explicitly available. Among others, map specifications that define quality criteria for a generalization task is essential. Whether or not initial data should be referred to in the quality evaluation, and how the data enrichment and data matching units are used are determined by map specifications. This is because the characteristics, relationships and patterns that should be preserved for a certain map purpose are defined in the specifications (see also Section 3.2). This research is motivated by general-purpose generalization of topographic data (Section 1.1.2) so NMA map specifications defines the purpose.

The three passages from existing data to quality evaluation as depicted in Fig. 3.1 are explained respectively in Sections 3.1.1–3.1.3. These three passages show an increasing degree of complexity in the evaluation and

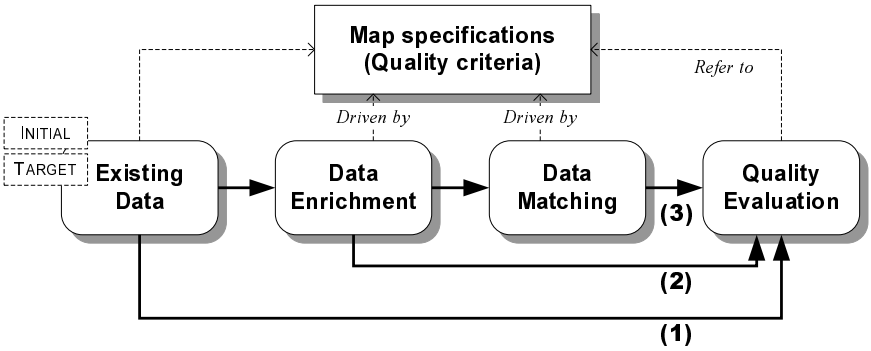


Figure 3.1 Overview of the conceptual framework for automated evaluation of generalization output

this thesis focuses on the most comprehensive approach (i.e. the evaluation following passage three).

3.1.1 *Evaluation based on existing data*[1]

The first passage is the easiest approach to automated evaluation. It depicts the cases where all required information is already available in the existing data. This usually happens in evaluating data against constraints on basic properties or relationships. Examples include size information encoded as attribute and topological relationships in a topological data model such as planar/network topology implemented in ESRI's Arc/Info and the 'formal data structure' (FDS) theory of Molenaar [133]. Note that a quite recent trend in industry is to calculate topological relationships on-demand rather than store them off-line. A notable example of this is the change of ESRI's spatial data model from topology-based (Arc/Info database) to object-relational (Geodatabases).

3.1.2 *Evaluation with data enrichment*

When map specifications require more information than existing data contain, the data have to be enriched. This is usually the case when a reasonable set of cartographic constraints is defined for the generalization. Since it is not possible to store all kinds of information in databases (some of which are task dependent while others are implicated in the locations and semantics of objects), implicit information (e.g. relative positions) should be calculated

[1] This research does not address the quality evaluation of thematic mapping where conceptual (feature class) generalization [see e.g. 111, 209] occurs.

on demand in order to enable automated evaluation of related constraints. The second passage in Fig. 3.1 describes this approach of evaluation. That is, existing data have to be enriched before can be evaluated. The evaluation of most cartographic constraints follows this passage, for example, constraints on distance, shape, orientation, density properties. Besides the properties and relationships, entity-level information should also be enriched in order to evaluate some constraints. Examples of implicit entities include entities that are aggregations or functional wholes of individual objects (e.g. clusters and alignments, networks and urban/commercial/rurual areas) and entities that are parts of other entities (e.g. a peninsula is a piece of land and an estuary is the wide part of a river).

3.1.3 *Evaluation with data enrichment and data matching*

The third passage is the most comprehensive approach to automated evaluation. It includes all the components in the second passage with an added component for data matching. This approach integrates the evaluation of both readability and preservation constraints. To automatically evaluate preservation constraints, target states of data should be examined to see whether they preserve initial states, and hence correspondences in both initial and target data should be linked [42, 44, 192]. The link should be established at individual, group and feature class level to enable automated evaluation at these levels. This approach is further explained in Section 3.5.

3.2 Analysis of map specifications

Map specifications are used to express the goals of generalization and to specify criteria for quality evaluation, which steer the evaluation processes. In Section 2.3.2 we outline previous effort to express map specifications in cartographic constraints. Most of the previous work has been dedicated to elaborating constraints for automated generalization. In this section, we describe our formulation of constraints for automated evaluation, which is slightly different from the constraints for generalization.

Section 3.2.1 distinguishes between readability and preservation constraints and discusses different ways in which they are to be evaluated. Section 3.2.2 describes our choice to divide the subject of constraints into three different levels of analysis. In Section 3.2.3 we give an in-depth analysis of the differences between entity, property, and relation level constraint and discuss the difference in evaluating different type of constraints. Finally, syntactic structure of constraints is analyzed which is insightful for further formalizing the constraints (Section 3.2.4).

3.2.1 *Readability and preservation constraints*

In this research we stick to the typology of constraints proposed in the EuroSDR project [44, 192] (see Chapter 2 for the basics of cartographic constraints). Cartographic constraints can be classified along two dimensions. The first dimension of classification is to distinguish constraints into legibility (readability) constraints and preservation constraints. According to Burghardt et al. [44], preservation constraints are satisfied before generalization (i.e. in initial data), while readability constraints are violated when changing the scale of the map and applying symbolization. When we try to resolve violations to readability constraints, preservation constraints are bound to be violated. From this point of view, readability and preservation constraints are two types of competitive design goals in generalization processes. It is almost never possible to fulfil both types of constraints for 100%.

Evaluation of readability and preservation constraints

Readability and preservation constraints are evaluated in different ways. The evaluation of readability constraints for the target data can be carried out independently of the initial data. The evaluation of preservation constraints, on the other hand, is always carried out by comparing the target data with the initial data [42, 44]. This requires to maintain or build corresponding relationships (links) between identical entities at different scales. The links can be established by generalization or data matching.

3.2.2 *Constraints at different levels of analysis*

The constraints can be classified along another dimension, that is, how many objects are involved in a constraint. We thereby distinguish between constraints for individuals (i.e. for one object and between two objects), for groups of objects and for feature classes. This in essence calls for a distinction of spatial entities in micro-, meso-, and macro-level entities for different levels of analysis. Ruas [163] for instance discussed the role of meso-level entities in automated generalization. A meso-entity could be a block of buildings, a cluster of buildings or a river network. Note that this research does not consider constraints on areal features that form a spatial partition of the mapped area (e.g. conceptual generalization of land-use registries) although the spatial partition is also a meso-/macro-level entity.

It is important to differentiate a constraint defined for individuals and aggregated for a feature class, from a constraint defined for a feature class (macro-level entity). In the former case, the properties to be respected (e.g.

size, shape and orientation) are measured from individual objects, and the aggregation (Section 3.3.3) aims to calculate an average degree to which the feature class violates the individual constraint. In the latter case, the properties to be respected are collective properties of the feature class (e.g. density and spatial distribution) which cannot be measured from the class objects individually.

3.2.3 *Entity-level preservation constraints*

Considering the constrained property in the structured template of constraints in the EuroSDR project (see Fig. 2.5), this research distinguishes between constrained property of entity-level, property-level and relation-level. In understanding their differences (see the list below), it is important to notice that an object is a micro (individual) level entity, a group of objects is a meso (group) level entity, and a feature class of objects is a macro (feature class) level entity (see Section 4.1.1 for a formal and detailed account).

- Entity-level: when the entity class (or entity classes) described in a constraint (see also Section 3.2.4) is constrained, i.e., the constrained property is entity class(es). Examples are 'important buildings should be kept' or 'alignments should be kept', where the constrained properties are the entity class 'building' and 'alignment'. Note that an alignment of objects is defined as a meso-level entity in this research rather than a relationship (see Section 4.1.4). Although an alignment describes a relationship among a group of objects, it has collective characteristics such as orientation and homogeneity and behaves as an entity at scale transitions;

- Property-/Relation-level: when a property of, or a relationship between, the entity class(es) is constrained in a map specification. Examples are 'target building shape should be similar to initial shape', 'minimum distance between buildings should be large enough' and 'target alignment orientation should be similar to initial orientation'. In these examples, the constrained properties are 'shape' property of buildings, 'distance' relationship between buildings, and 'orientation' property of alignments.

While property-/relation-level constrained property can occur on both readability and preservation constraints, it appears that entity-level constrained property does not exist in readability constraints. This research therefore refers to the latter as entity-level preservation constraints. Furthermore, entity-level preservation usually implies that the entity class(es) of

the constraint is important or of higher priority and hence should be kept completely in the generalization process.

Evaluation of entity-level preservation constraints

Entity-level preservation constraints are evaluated in a way that differs from property-level preservation constraints, although both types of preservation constraints should be evaluated by comparing the target data with the initial data (see Section 3.2.1).

Property-level preservation implements the comparison by checking the states of the entities that are kept in the target data and referring back to their initial states. It does not penalize the deletion operation. On the contrary, the evaluation of entity-level preservation works in a reverse direction: the comparison starts from the initial data and checks if all the initial entities of a feature class are still represented in the target data. Any deletion of an entity in the class will be penalized by this type of constraint.

The differences in the two types of preservation which lead to the difference in their evaluation are further explained here. First, one may argue that a property-level preservation automatically implies a entity-level preservation. This statement is however inaccurate. We would articulate more precisely the property-level preservation as: for those entities that have been kept in the target data, their constrained properties should be preserved (e.g. building shape should be preserved). Since the property-level preservation only respects the properties of those kept entities rather than how many initial entities are kept, deletions of initial entities do not violate this type of constraints [42].

Entity-level preservation constraints, on the other hand, imply that ***ALL*** entities of a certain type in the initial data (e.g. all important buildings or all alignments in above-mentioned constraints) should still be represented in the target data, and should not be deleted. This statement holds also for meso-/macro-level entities. For example, although the composing objects of an alignment may be deleted, it is still true to say that the entity specified in the constraint on keeping alignments is not deleted, since this alignment is the (meso-level) entity in the constraint rather than are its composing objects. On the other hand, while entity-level preservation does not allow deletion of entities, it is important to note that this type of constraint may still be characterized as soft constraints (Section 2.3.2), meaning that they can be relaxed to meet other more important constraints (Section 3.3.1).

To summarize, deletions of entities do not violate property-level constraints but they do violate entity-level preservation constraints. If one only uses property-level constraints, the evaluation results may be misleading. For example, if all initial buildings were deleted during the generalization, the evaluation of the target data will report no violation to e.g. the shape preservation constraint. This is because there is no building in the target data which can violate this constraint (no penalty principle). Given this fact, it seems to be not the case for the constraints that preserve the spatial distribution of a feature class of objects. With the deletion of objects in a feature class, the constraint on spatial distribution is bound to be violated; if all objects were deleted, this property-level preservation is completely violated. However, one should note that the entity described in this particular constraint is the feature class (macro-level entity) rather than the composing objects (mirco-level entity). 'Object being deleted' is different from 'entity being deleted'. A feature class entity will never be deleted unless the class definition is removed from the database scheme (see also the formal description of feature class in Section 4.1.1). If a feature class entity is deleted (e.g. building class is not modeled at small scales), constraints on this class are also obsolete. From this point of view, it still holds that deletion of entities do not violate property-level constraints for feature class level entities. Conversely, any deletion of an entity violates the entity-level preservation constraint. If all initial entities were deleted, the entity-level preservation constraint (e.g. important buildings should be kept) is completely violated. In this sense, entity-level preservation serves as a safeguard of property-level preservation. In this research, we give an example of entity-level preservation on building alignments and its evaluation in Section 6.4.

3.2.4 *Syntactic structure of individual constraints*

In this section, we use linguistic structures to analyze the structural aspects of cartographic constraints. Syntactically, a cartographic constraint can be viewed as a statement consisting of *subject*, *subject modifier*, *predicate*, *object*, and *object modifier* [236], where the authors have also provided several examples in which constraints are reformatted using this syntactic approach. The advantage of the syntactic analysis is that it decomposes the knowledge embedded in cartographic constraints into a small number of building blocks (syntactic terms), and that by combining the syntactic terms and their subtypes we can describe almost all kinds of constraints. However, since the use of the syntactic terms such as *subject* and *object* can be confusing in the context of constraint expression, this section uses more common terms to express the same syntactic structure. Some of the terms are brought from the constraint structure described in [190, 192]. The terms

are *entity class*, *scope modifier*, *constrained property*, and *condition*. In the following, we first explain each of these terms, and then we describe how various kinds of cartographic constraints can be composed using the syntactic structure.

Entity class

Entity class is the spatial entities to be considered in a constraint (the same to the syntactic term subject/object in [236]). Syntactically, a constraint may aim at a single entity class (e.g. building polygons should be large enough), or it may aim at multiple related entity classes (e.g. distance between buildings and roads should be large enough). For preservation constraints, entity classes at different scales should be related. Entity class is a much broader concept than feature class. In this research, entity class is divided into individual object, group, and feature class of objects, each of which can be further distinguished in terms of feature class and geometry type (see Table 3.1 for example).

Scope modifier

Scope modifier is a specification to limit the scope of the entity class(es) to which a constraint may apply (the same as the modifier term in [236]). In other words, a scope modifier is a condition to define the subset of the entity class to be considered in a constraint. First, scope modifier could be semantic, specifying the subtype (e.g. built-up area) of a feature class (e.g. land-use) to be considered by a constraint. Second, scope modifier could be conditional, filtering out those entities that do not satisfy some condition(s). Third, scope modifier could be contextual, limiting the scope of a constraint to some spatial context (see Table 3.1 for example). As to the types of spatial context, we refer to the classification proposed by Mustière and Moulin [140].

Constrained property

Constrained property is the property of entity class(es) to be constrained by a constraint. For instance, in the expression 'building polygons should be larger enough', 'buildings' are the entity class and 'size' is the constrained property. According to Section 3.2.3, there are entity-level, property-level, and relation-level constrained property (see Table 3.1 for example).

Syntactic terms	Types	Extra dimensions/Subtypes	Examples
Entity class	Individual-level		building polygons; road centerlines
	Group-level	Feature Class Geometry Type	building alignments; hydrography networks
	Class-level		building polygon layer; land parcel layer
Scope modifier	Semantic		'high-rise' buildings
	Conditional		*Size*([initial]) $\geq \sigma$
		in significant group	buildings of a cluster
	Contextual	in particular area	houses in rural area
		in relation to other objects	hotels nearby railway stations
Constrained property	Entity-level		important building to be kept
	Property-level		minimum size
	Relation-level		minimum distance
Condition	Expression		P([target]) or P([initial], [target]){>, =, <}μ
	Rule		If [initial] Then [target]

Table 3.1 Syntactic analysis of cartographic constraints

Condition

Condition to be respected specifies the conditions to be respected for the entity class(es) in a constraint. Conditions can be expressed in two general forms, namely the conditional expression and the rule. A conditional expression is a mathematic expression with relational operators, e.g., $\{>, =, <\}$. A rule is usually a `if-then-else` statement (see Table 3.1 for example). Note that condition and constrained property together form the predicate term described in [236].

Express constraints with the syntactic structure

The simplest kinds of constraints can be combined by $\langle$ *entity class*, *constrained property*, *condition* $\rangle$, e.g., 'target building should be larger than a threshold'. The most complex ones can be combined by $\langle$ *entity class*, *scope modifier*, *entity class*, *scope modifier*, *constrained property*, *condition* $\rangle$. An example of this is 'the road leading to a building in a peninsula should be kept'. This is a preservation constraint where the first entity class is roads in the initial data, the scope of the class is the roads leading to buildings, the second entity class is buildings in the initial data, the scope of the class is the buildings located in peninsulas, constrained property is roads and buildings (i.e. entity-level), condition to be respected is that the buildings and roads should exist in the target data.

Combining the syntactic terms and their subtypes can result in a huge number of different kinds of constraints. Let us consider a topographic data model consisting of nine feature classes (Administration, Building, Railway, Road, Relief, Lake, River, Coastal feature, Land-use) and of three geometry types (point, linear and areal) and the syntactic structure in Table 3.1, the following estimations apply. For a simple combination, it results in 2'430 kinds of constraints, i.e., 3 types of entity classes (individual-, group-, and class-level) $\times$ 9 feature classes $\times$ 3 types of geometries $\times$ 5 types of scope modifiers $\times$ 3 types of constrained properties $\times$ 2 forms of conditions. A similar calculation result in 984'150 kinds of constraints for the most complex combination.

The above estimations show the potential of using the syntactic structure to characterize various kinds of cartographic constraints. For example, all of the generic constraints collected in the EuroSDR project [190, 192] can be characterized using the proposed syntactic structure.

3.3 EVALUATION OF CARTOGRAPHIC CONSTRAINTS

This section discusses properties of cartographic constraints needed for automated evaluation, distinguishing between importance and priority of constraints (Section 3.3.1), presenting the concept of constraint violation or satisfaction (Section 3.3.2), and describing ways to present evaluation results (Section 3.3.3).

3.3.1 *Importance and priority of constraints*

The competitive nature of readability and preservation constraints (see also Section 3.2.1) makes that generalization output is an optimization or compromise between conflicting design goals (constraints). Consequently, quality evaluation of generalization output requires that constraints should be satisfied as much as possible, and that important constraints should be better respected than less important ones.

Importance is a key property of constraints for automated evaluation. It reflects the degree to which a constraint contribute or influence the quality judgment of the whole map ('AGGREGATION' in Section 3.3.3). Relative importance is usually determined by cartographic experts by specifying weights to different constraints. A constraint is regarded to be important probably because the constrained property (e.g. shape) or the involved feature class (road) is important for a certain application. Thus the relative importance of constraints can be altered in different situations or tasks.

Priority specifies the sequence in which the constraints should be resolved in generalization, which is different from the concept of importance. For example, constraints of higher priorities should be resolved first even though they are of less importance. This is because violations to some important constraints (e.g. non-overlapping) have to be resolved after some less important ones (e.g. density). As a result, priority is a concept highly relevant to the generalization process, where a set of operations should be specified to resolve several cartographic conflicts in sequence. The concept is nevertheless irrelevant to quality evaluation of generalization output, as the evaluation only checks the final states rather than the process, and thus the sequence in which the involved constraints are evaluated does not change the evaluation results (i.e. quality scores). Importance conversely refers to the fulfilment of the constraints in the end product.

3.3.2 *Violation or satisfaction of constraints*

The constraint violation (CV) reflects the degree to which a constraint is violated, and is derived from data by measuring the constrained property of an entity in relation to its initial states or against map specifications. In this research, CV is scale into $[0, 1]$, with zero indicating no violation and one indicating maximum violation to the constraint for the subject (an individual, a group, or a feature class) to be considered. The constraint satisfaction (CS) is opposite to CV: $CS = 1 - CV$. Both values can be used to describe the quality of generalized maps.

Note that threshold based constraints almost always yields boolean output (CV = 0 or 1). A study on interactively generalized data shows that a value (e.g. size) that is slightly below the threshold is usually considered to be acceptable [42, 192]. This implies that a degree of flexibility is always introduced in manual or interactive generalization. For automated evaluation, a tolerance value should also be considered together with threshold to account for the flexibility, and to convert the boolean output of the threshold based constraint into violation degrees from zero to one. This means in fact that some constraint violations can be a fuzzy measure in stead of a binary measure.

Furthermore, the continuous range of CV or CS can be qualified into ordinal values, such as good, nearly good, nearly bad and bad acceptability. This is much easier for human to interpret the evaluation results but the qualification process may also be problematic.

3.3.3 *Presentation of evaluation results*

The quality of generalization output can be presented in many different ways. First, individual assessments can be aggregated into a single value to reflect the overall quality. Second, the constraint violation or satisfaction can be presented graphically. Third, quality of generalized datasets can be presented on maps where cartographic errors can be visualized.

Aggregation of constraints and cost function

The evaluation of individual level, group level and feature class level constraints results in individual assessments (in the form of CV or CS). Usually the individual assessments can be aggregated to give a meaningful description of the generalized data/map. The individual quality assessments can be aggregated at several levels. At entity level, the assessments of different properties (size, shape, orientation, etc.) of an entity can be aggregated

to reflect the general quality of this generalized entity. Assessments for individuals and groups of objects can be aggregated for a feature class to reflect the average violation or satisfaction of the class to particular constraints. For example, see to what extend the shape preservation constraint is kept for building feature class. Note that the evaluation of feature class level constraints need not be aggregated, unless one wants to summarize over several classes (minimum sizes for building and vegetation classes) or constraints (preservation of shape and orientation of buildings). Finally, constraints at different levels and for different purposes can be aggregated for a whole map, to reflect the general quality of the whole map with respect to the defined constraints. Different aggregation strategies have been elaborated in [19, 21, 120].

To mathematically define the aggregation, relationships among constraints should be considered. Currently, the most practical approach to the aggregation is via the so-called **cost function** in which the inter-relationship of constraints is ordered linearly. Its general form is defined as follows:

$$Cost = \sum_{i=1}^{n} CV_i \times Weight_i \Big/ \sum_{i=1}^{n} Weight_i \tag{3.1}$$

where $Weight_i$ indicates the relative importance of CV_i and is determined (by users) depending on specific contexts. The cost function can be applied to different levels of aggregation and its output is scaled to $[0, 1]$. A lower value indicates a less violation (or better quality) of the involved constraint(s), and vice versa. Note that one can develop a satisfaction function similarly by replacing CV with CS. Besides, the qualification described in Section 3.3.2 applies to both cases. This approach has been used in previous research on automated evaluation [see e.g. 19, 21, 85, 144]. The limitation of this approach is that it assumes that different CVs can be linearly combined. The next sections describe ways to present evaluation results if such a combination approach is problematic.

Visualization of constraints

Sometimes viewing the evaluation results per constraints graphically provides more insights into different quality aspects of the generalized data. Bar chart [e.g. 73, 105] and parallel coordinate plots [e.g. 144] for instance are used to visualize violations or satisfactions to individual constraints (see Fig. 3.2 for an example of the bar chart view of CVs).

Visualization of evaluated data/maps

Sometimes the numerical evaluation resulted from e.g. a cost function may not be sufficient. Hence visualization of evaluated maps is necessary for human to interactively inspect the quality, for example to tune evaluation parameters for different areas. Usually the quality indicator (CV or CS) can be attached to entities (individuals or groups) as an attribute value and visualized. If the map is too large, visualizing the whole map does not make too much sense. In that case, sampling of different areas of different contexts can be useful.

3.4 Balanced map specifications

This section proposes guidelines for formulating a set of evenly balanced cartographic constraints (Section 3.4.2). A motivational example in Section 3.4.1 shows that a set of unbalanced constraints would however give misleading (over-optimistic) evaluation results. This example also motivates the topics (constraints) that are discussed in Chapters 5 and 6.

3.4.1 *A motivational example*

As discussed previously (Section 1.2), a meaningful evaluation requires a set of evenly balanced constraints [88, 192], and a set of unbalanced constraints may lead to over-optimistic evaluation results [88]. An example is shown in Fig. 3.2 to illustrate this idea.

In Fig. 3.2, an initial dataset at 1:25k (from ICC) and three generalization outputs are displayed. Hypothetical constraint violations are estimated for these outputs with respect to the set of constraints described in Fig. 3.2. The first three constraints are readability constraints on one object and between two objects; the latter two constraints are group or feature class level preservation constraints. We assume that output (b) is the optimal solution amongst the three, and output (d) is the one that with the worst acceptability.

Numerical evaluation results, i.e., constraint violations, may yield misleading conclusions if the set of constraints is inappropriately selected. If one select a set of constraints that only consists of readability constraints (e.g. the first three constraint in Fig. 3.2), the evaluation would say that output (d) is the best solution amongst the three, and output (b) is the worst. However, output (d) was obviously a bad generalization because all buildings were deleted. The three readability constraints are regarded to be not violated at all due to the way how CV or CS is calculated for

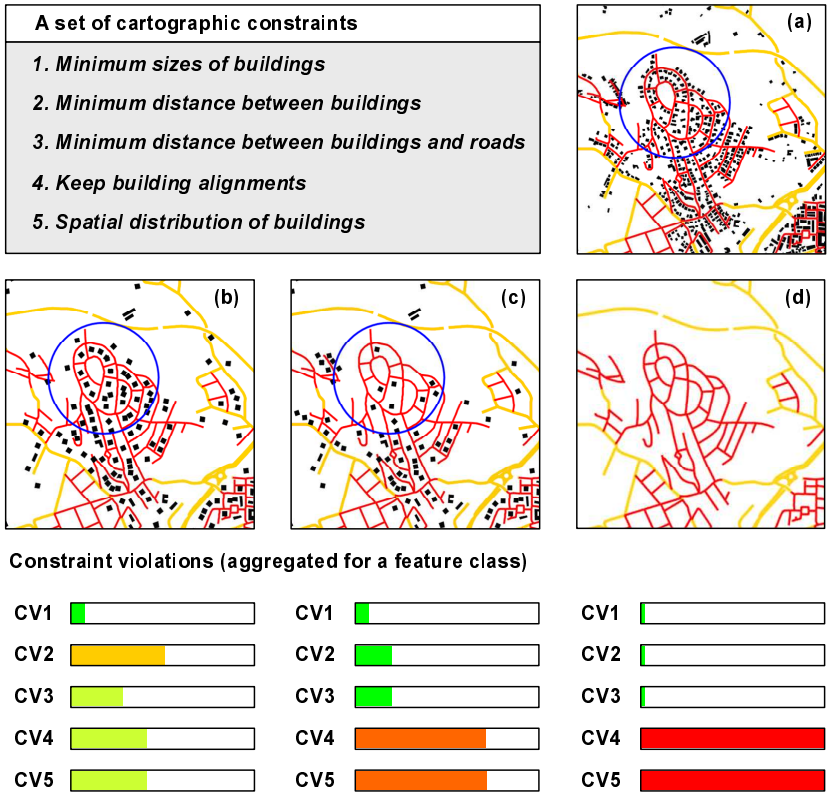


Figure 3.2 Hypothetical evaluation against a set of constraints. Top: the constraint set and the initial dataset (a); middle: three generalization outputs (b)-(d); bottom: bar chart of hypothetical constraint violations (green indicates low severity and red indicates high severity)

readability constraints (see Section 3.2.1). A related idea is discussed in Section 3.2.3 on the evaluation of entity-level and property-level constraints.

Burghardt and Schmid [42] identified a similar evaluation situation and they came to the conclusion that individual assessments need a careful interpretation. I agree with this conclusion and add that a selection of a more appropriate set of constraints should better account for it. For example, by adding two preservation constraint for groups and feature classes, we obtain a better view of the generalization quality. That is, by making a compromise between the five constraint in Fig. 3.2, output (b) is evaluated as the best solution, output (c) the second, and output (d) the worst. This

result is consistent with our assumption about the generalization quality.

3.4.2 *Guidelines for a meaningful evaluation*

A fundamental principle of constraint selection for a meaningful evaluation is to select constraints from conflicting design goals (e.g. readability vs. preservation), so that the conflicting goals can be optimized (or compromised) when making evaluation conclusions based on the set of constraints. The proposed principles are purely empirical and hence serves only as guidelines but not as rules. More specific guidelines on the selection of a set of evenly balanced constraints are described as follows:

- When a readability constraint (e.g. minimum size of buildings) is selected, a related preservation constraint (e.g. maintain building sizes) should be selected. This can avoid any of the constraints being overcooked in the generalization process;

- Entity-level preservation constraints, if available, should be selected to avoid the over-optimistic evaluation of property- or relation-level constraints due to deletion of entities;

- Group or feature class level preservation constraints should be selected in addition to individual level constraints.

Note that group or feature class level preservation constraints are used more generally as a way to avoid the over-optimistic evaluation due to deletion of entities. Group level constraints are used to retain important patterns and structures on the maps. An example of evaluating entity-level versus property-level preservation constraints on building alignments is shown in Section 6.4. The constraints that preserve the spatial distributions of feature classes is generally required to ensure that a simplified representation has a similar spatial distribution to the initial distribution. This is also crucial for keeping spatial structures and characteristics at a global level.

Based on the above principles, a set of constraints related to building feature class is exemplified in Table 3.2. This further justifies the major focus of this research on formalizing and measuring constraints on building alignments and spatial distribution (data enrichment), and on matching corresponding entities at different scales for automated evaluation of preservation constraints.

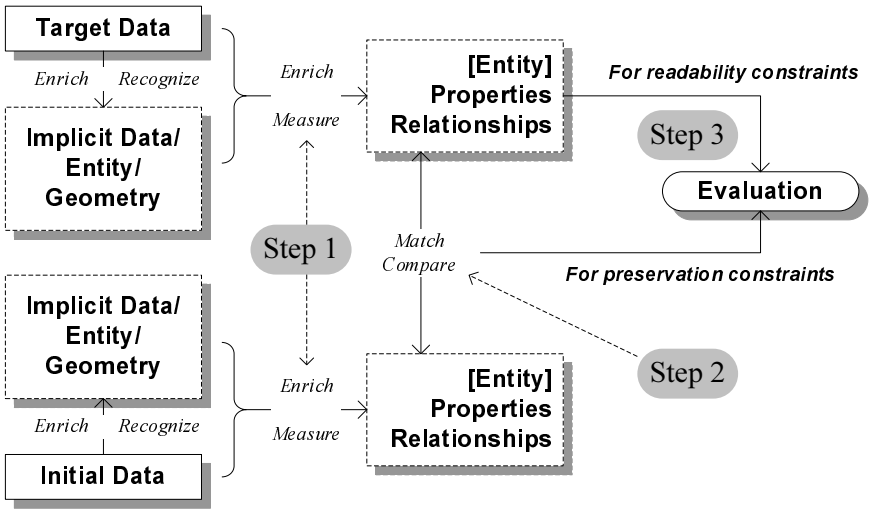


Figure 3.3 A three-step approach to automated evaluation

3.5 A THREE-STEP APPROACH TO AUTOMATED EVALUATION

In addition to the requirement for a set of balanced constraints to draw meaningful evaluation conclusions, there are also requirements for system components to enable automated evaluation. Based on the assumption that an enabled automated evaluation system should be equipped with facilities to enrich, match and evaluate topographic data, we propose a more detailed three-step approach (i.e. data enrichment, data matching, and evaluation in Fig. 3.3) that refines the evaluation framework described in Section 3.1.

To begin with, the approach enriches existing data with information re-

	Readability	Preservation
Individual	minimum dimensions minimum distance	shape relative position
Group		alignment shape; density
Feature class		density spatial distribution

Table 3.2 An example set of balanced constraints for building class

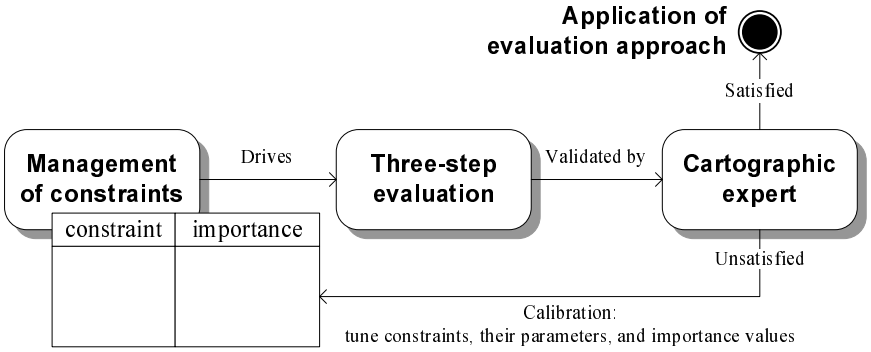


Figure 3.4 Calibration for the proposed evaluation approach

quired by the constraints to be evaluated (Step 1 in Fig. 3.3). The data enrichment is driven by the specific set of constraints used and consists of two stages. First, if the spatial concepts (e.g. semi-detached houses and alignments) described in a constraint is not explicitly available, we enrich the data with explicit entities (or geometries) of the concept. Such concepts are usually higher-level semantics, functional wholes (groups) or parts of existing data. As a result, data enrichment in this stage is a feature detection or pattern recognition process. The second stage enriches the constrained property (e.g. shape, density, and spatial distribution) specified in a constraint if unavailable. Data enrichment in this stage is a measuring process, where properties and relationships of both existing and enriched data entities and geometries are measured.

If the constraint to be evaluated is a readability constraint, that is, its evaluation is independent of the initial dataset, automated evaluation is carried out by firstly enriching the target dataset and then assessing the measured property or relationship with respect to the condition of the constraint. If the constraint to be evaluated depends on the initial dataset (e.g. a preservation constraint), both the initial and target datasets should be enriched (recognized and measured). After that, corresponding spatial entities in the initial and target data should be matched (Step 2 in Fig. 3.3). Finally, the measured property of or relationship between linked entities can be compared and evaluated against the condition defined in the constraint (Step 3 in Fig. 3.3).

To be practically useful, there should be an (inner) feedback loop performed interactively by cartographic experts to calibrate the proposed evaluation

approach before the approach is used (see e.g. Fig. 3.4). In the calibrating loop the experts could tune the constraints used, their parameters, and relative importance values until a combination of visual and numerical evaluation is considered satisfactory by the experts. Such a calibration process is needed in practice (e.g. production lines) [120] but is not discussed further in this thesis.

3.6 Summary

This chapter presents a conceptual framework for automated evaluation, which highlights the requirement for explicit modeling of information to facilitate automated processes. This includes the formalization of cartographic constraints, enrichment of topographic data, and data matching to link corresponding entities at different scales. To begin with, we analyze aspects (including the syntactic structure) of cartographic constraints, which provides a starting point for constraint formalization. Then, we elaborate an example to demonstrate the idea that the evaluation based on unbalanced map specifications may lead to misleading conclusions. We thereby propose empirical guidelines on the selection of a set of evenly balanced constraints for a meaningful evaluation. Finally, the idea of balanced map specifications and how they should be evaluated provide insight into the proposed three-step approach to automated evaluation of generalization output (i.e. data enrichment, data matching, and evaluation).

The conceptual framework will be further developed at different levels of realizations. In Chapter 4, we develop a theory to formally capture aspects of the conceptual framework, in which cartographic constraints can be unambiguously defined. In Chapters 5 and 6, components of the evaluation framework (data enrichment, matching, and evaluation) are designed and implemented. As we point out in Section 1.3.1, building and road feature classes are used as a proof-of-concept in the design and implementation phases.

4 A formal theory for automated constraint-based evaluation

This chapter presents a theory for formalizing aspects of automated evaluation framework described in Chapter 3 and especially the cartographic constraints in a first-order predicate representation. The theory is intended for an unambiguous definition of cartographic constraints for quality evaluation of multi-scale data. Although the formal theory is targeted at a specific application, i.e., the generalization evaluation, it also captures generic aspects of, and thus is applicable to, the geospatial domain in general. This chapter is closely related to the research questions regarding 'how to formalize cartographic constraints and complex concepts and relationships thereof?' and 'which formal concepts are useful to precisely define map specifications for automated evaluation?'.

This chapter firstly presents an object-oriented (OO) model for spatial entities and relationships of MRDBs, with semantics of the model defined and modeling principles discussed (Section 4.1), where significant patterns (groups) and complex relationships are exemplified. Then Section 4.2 gives formal descriptions to the specification of characterization, evaluation, and aggregation functions. On top of the entity-relation model, cartographic constraints are formalized into first-order logic expressions (Section 4.3). Finally, Section 4.4 discusses the expressive power of the theory, semantics of spatial relationships, possibilities for automated evaluation, and implementation issues.

OWN PUBLICATIONS

This chapter is partly based on the following publications:

- X. Zhang, J. Stoter, and T. Ai. Formalization and automatic interpretation of map requirements. In *The 11th ICA Workshop on Generalisation and Multiple Representation*, page 12, Montpellier, France, 2008.

- X. Zhang et al. A theory for cartographic constraint formalization. In *(SDH2012)*, Bonn, 2012 (submitted manuscript).

4.1 Spatial entities and relationships

Entities, objects, groups, and feature classes are used extensively in describing the conceptual framework for automated evaluation in Chapter 3 and especially in the discussion of entity-level preservation constraints (Section 3.2.3). However, their intended meanings and the relationship between the concepts are not clear and hence should be further clarified. An ontological distinction between different categories of entities helps to clarify the terms used and, since the terms are also extensively used in expressing cartographic constraints [e.g. 1, 192], the clarification is also necessitated by the need to precisely define cartographic constraints for automated processes. Besides, an OO approach is used to capture aspects of a spatial data model where entities and relationships can be easily expressed.

This section distinguishes between three categories of entities: individuals (objects), universals (classes), and collections (groups), according to the findings in theories of ontology research [e.g. 30, 122, 123, 181, 182]. This ontological distinction also supports the conventional distinction between micro-, meso-, and macro-level entities made in the generalization domain. For example, a map object (micro-level entity) is an individual, a group of objects (meso-level entity) is a collection, and a feature class is a universal. In the following, I give formal definitions of the entities in the geospatial context (Section 4.1.1) and explicate the relationships between different categories (Section 4.1.2) and spatial entities (Section 4.1.3). The purpose of this section is not to present a full-fledged spatial data model but to sketch out a formal theory for describing the evaluation framework. Hence only related aspects of a data model are presented. Several examples applying the formal theory to real-world cases are shown in Section 4.1.4.

4.1.1 *Entities, objects and groups*

This research identifies entity[1] as an abstraction which constitutes the basic unit of manipulation in the formal theory for the evaluation process. An entity in this research can be a data object (feature) or an abstract object (group, semantic object, or feature class), see their relationships in Fig. 4.1. It results from an OO design and can be formally given by a 2-tuple:

[1] The term entity has exactly the same meaning as the entity in 'entity-level preservation constraint' in Section 3.2.3

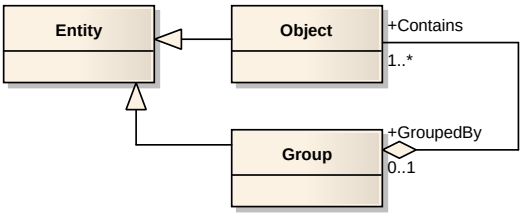


Figure 4.1 UML diagram of entity, object, group and their relationships

$$\mathrm{Entity} = \langle \mathrm{ID},\ \mathrm{Granu} \rangle \tag{4.1}$$

where:

- ID is an integer representing an unique identifier of the entity (object or group) being referred to;
- $\mathrm{Granu} \in \{micro,\ meso,\ macro\}$ indicates the granularity of the entity.

Granularity and scale (LOD) are related concepts in the generalization domain, but they have slightly different meanings. Scale describes the ratio of a distance on the map to the corresponding distance on the ground which usually indicates the levels of detail to be represented on the map; whereas granularity specifies the degree to which objects are grouped. For example, a building object is a micro entity, a partition or an alignment of objects is a meso entity, and a feature class of objects is a macro entity. Such a division is a convention in the generalization domain (see e.g. [22, 162, 163]), which is also similar to granular partitions as described in [29].

Among others, **object** and **group of objects** are two fundamental concepts derived from entity. In the following, formal definitions of object, group and feature class are given.

An **object** is an individual which can be formally given by a triple:

$$\mathrm{O} = \langle \mathrm{ID},\ \mathrm{CL},\ \mathrm{Geo} \rangle \tag{4.2}$$

where:

- CL is the feature class type to which the object belongs, e.g. Building or Road;
- Geo is the geometry of the object, which is composed of: (i) the type of geometry $\mathrm{Geo.type} \in \{point,\ linear,\ areal\}$ and (ii) the spatial

representation (coordinate pairs) $Geo.rep = \{p_1, \ldots, p_n\}$; $n > 1$ for linear objects and $n > 2$ for areal objects.

A **group** is in nature a collection (set) of objects. For the purpose of this research, groups are identified as explicit entities with identifiers, because groups exhibit structural properties (e.g. density, spatial distribution and other statistical properties) which should be preserved during the generalization, and which mostly cannot simply be obtained by the sum of individual properties of group members. A group is given by a 4-tuple:

$$Gr = \langle ID, OC, R, Geo \rangle \tag{4.3}$$

where:

- $OC = \{O_1, \ldots, O_n\}, n \geq 0$ is a collection of objects that forms the group;
- R is the relationship between spatial objects; a group is constrained by some relationship R which holds among the group members;
- Geo has also the type and spatial representation components as described above. It is about the representative geometry of the group but not of its members. As a result, the geometry type of a group may be different from those of its members. For example, the representative geometry of a group can be the minimum bounding box (MBB), the convex hull, the skeleton, the centroid, etc. of its object collection OC, which is needed to facilitate the visualization, manipulation and analysis of the group.

This definition is flexible enough to cover various kinds of group entities. First, a group can be a collection where some relationship holds for its members (e.g. alignment relationship), or between the members and some other objects (e.g. neighborhood and partition relationship). Second, the above definition also allows groups of objects at different scales, or from different feature classes that create 'higher-level' phenomena, e.g., a 'housing aggregate' composed of a house, a garden and an outbuilding (garage or annex). The group entity in the second sense is similar to the concept of *composite object* described in [209]. In describing group entities in the proposed evaluation framework, R is restricted to the relationships between objects within the same scale (i.e. *intra-scale* relationships). Inter- and intra-scale relationships will be discussed in detail later in Section 4.1.3.

Feature class in the geospatial domain is a categorical concept (corresponding to kind or universal in e.g. [30, 182]). A feature class only contains type information and is given by $\mathtt{FeaCls} = \langle \mathtt{Cls}, \mathtt{Geotype} \rangle$, where $\mathtt{Cls} \in \{\mathtt{Building}, \mathtt{Road}, \ldots\}$ and $\mathtt{Geotype} \in \{\mathtt{point}, \mathtt{linear}, \mathtt{areal}\}$. Note that in the proposed approach, I distinguish different classes also depending on geometry types, as geometry types are important information which could better formalize map specifications for automated processes. This treatment is similar to data models in modern spatial databases (e.g. ESRI Geodatabase, Oracle Spatial and Postgres).

The semantic relationship describing that an object is an instance of a feature class can be defined as:

$$\begin{aligned} \mathit{InstanceOf}(\mathrm{O_i}, \mathtt{FeaCls_i}) \equiv (\mathrm{O_i}.\mathrm{CL} = \mathtt{FeaCls_i}.\mathtt{Cls}) \wedge \\ (\mathrm{O_i}.\mathrm{Geo}.\mathtt{type} = \mathtt{FeaCls_i}.\mathtt{Geotype}) \end{aligned} \tag{4.4}$$

Now, all instances of a feature class C can be expressed as a specialized group using combined constructs from Def. 4.3 and 4.4:

$$\mathrm{Gr_C} = \langle \mathrm{ID}, \{\mathrm{O_i} | (\forall \mathrm{O_i}) (\mathit{InstanceOf}(\mathrm{O_i}, \mathrm{C}))\}, \emptyset, \mathrm{Geo} \rangle \tag{4.5}$$

Here the object collection ($\mathrm{Gr_C}.\mathrm{OC}$) is specialized using a set-builder notation, meaning that for all objects in the domain of discourse, any thing that is an instance of the class C is an element of $\mathrm{Gr_C}.\mathrm{OC}$. This group $\mathrm{Gr_C}$ actually means the extension of the feature class C in the universe. In a specific application, $\mathrm{Gr_C}$ can be used to indicate all the instances of a feature class that exist in a particular dataset, which also means that $\mathrm{Gr_C}.\mathrm{OC}$ could be an empty set if all the instances of the class have been deleted from the dataset. This means that a feature class level entity always exists no matter it contains member objects or not, as a feature class entity is defined in database scheme (e.g. as a table structure). This support my argument about property-level preservation constraint for feature class entities in Section 3.2.3.

Note that, it would be ambiguous to specify all instances of a feature class in this form: $\langle \mathrm{ID}, \mathrm{OC}, \mathit{InstanceOf}, \mathrm{Geo} \rangle$, which only states that the members of a group (e.g. a cluster of buildings) are instance of a class (e.g. Building), but not state that all instances of the class are members of the group. Besides, specifying explicitly the relationship *InstanceOf* is not necessary, because it can be inferred from the group members' class type ($\mathrm{O_i}.\mathrm{CL}$). So in Def. 4.5 $\mathrm{Gr_C}.\mathrm{R}$ is set to $\emptyset$. To syntactically distinguish a meso-group from

	Individual	*Universal*	*Collection*
Individual	*part-of*	*instance-of*	*member-of*
Universal		*subsumption (is-a); partonomic inclusion*	*subset-of; partonomic inclusion*
Collection	*partition-of*	*extension-of*	

Table 4.1 Ontological relationships between categories, from [30]

a macro-group (i.e. a feature class of objects), a non-empty relationship R should be explicitly specified for a meso-group to explain how the group members are related (see examples in Section 4.1.4).

To further distinguish different entities and the scale for which they were collected, E_1^{S1}, E_2^{S1} and E_3^{S2} for instance are used to denote two entities of the same scale (the former two) and another entity of a different scale. This notation system can also be applied to object, e.g., O_1^{S1}, and group of objects, e.g., Gr_1^{S1} for a meso-group and Gr_{C1}^{S2} for a macro-group (feature class).

4.1.2 *Ontological relationships between categories*

The fundamental ontological relationships between three categories of entities, i.e., individuals, universals and collections, are summarized in Table 4.1. In the following, I give some example of these ontological relationships, especially those closely related to automated generalization. Note however that, formal definitions to these ontological relationships is not given in this research because a rigorous treatment needs sophisticated comprehension in the study of topology and mereology [78, 180, 210], which is out of the scope of this research. The semantics of these relationships and the related theorems, for instance, have been given in an axiomatic system by Bittner et al. [30].

Subsumption or taxonomic inclusion (*is-a*) is a relationship between two universals (feature classes in this research). It is defined on top of the *instance-of* relationship. That is, class A *is-a* class B if every instance of A is a instance of B (e.g. Building *is-a* Construction or Deciduous Forest *is-a* Forest). The definition of *instance-of* is given before (Def. 4.4). In the generalization domain, taxonomic inclusion is usually used to form *classification hierarchy* of classes (e.g. of land-use and forest) which corresponds to conceptual

generalization [111, 199, 209].

Feature classes, or feature types, can be related in a classification hierarchy, so that the hierarchy forms a system of types and subtypes (see Fig. 4.2). According to the definition of feature class, all classes and subtypes in a hierarchy can be denoted using this template: $\langle <\textit{class_name}>, \mathtt{Geotype}\rangle$. For notational convenience, I express the fact that an object, O_i, is an instance of a class, $<\textit{class_name}>$, in a predicate template:

$$<\textit{class_name}>(O_i) \equiv \textit{InstanceOf}(O_i, <\textit{class_name}>) \tag{4.6}$$

where the placeholder $<\textit{class_name}>$ can be substituted by any feature class or subtype in an application domain. In the proposed evaluation framework, for example, $\textit{Building}(O_1)$ means that O_1 is a building and $\textit{Built_up_area}(O_2)$ means that O_2 is a land use parcel of a subtype 'built-up'. Given a class, its super-class/type and sub-classes/types can be identified in the classification hierarchy (e.g. Fig. 4.2).

Partonomic inclusion of universals (classes), on the other hand, is defined on top of the individual *part-of* relationship. That is, class A is partonomically included in class B, if and only if (i) every instance of A is an individual part of some instance of B, and (ii) for every instance y of B there is some instance x of A such that x is an individual part of y. The partonomic inclusion is used in the generalization domain to form *aggregation hierarchy* [203, 204].

Note that aggregation hierarchy of classes is by definition different from *object hierarchy* [133], because that the *part-of* relationship holds between two objects does not necessarily follow that the partonomic inclusion holds between their respective classes. For example, it is not true to say that the concept Garden is a partonomic inclusion of the concept Building Block just by seeing a building block that is composed of some houses and gardens (an object hierarchy). There may exist some other building blocks which do not

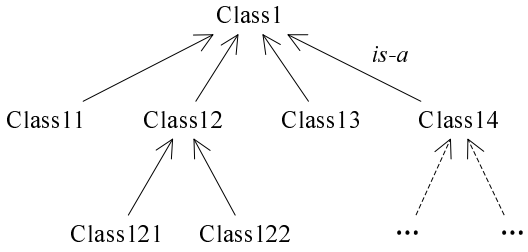


Figure 4.2 Schematic example of a classification hierarchy

contain any garden at all, or some other gardens which are not part of any building block (e.g. gardens in a natural park). Many implementations of generalization and multi-scale representation systems show a trend of using tree-based data structures to represent the classification and aggregation hierarchies [5, 9].

Likewise, partonomic inclusion can be defined for collections on top of the individual parthood between collection members. For example, the collection {*peninsulas of countries a, b and c*} is partonomically included in the collection {*countries a, b and c*}. The *subset-of* relationship between collections is, however, different from partonomic inclusion of collections. For example, an alignment of buildings is a subset of a larger cluster of buildings (see examples in Fig. 7.2(a)-7.2(c) of Section 7.2).

The following exemplifies some remaining relationships in Table 4.1 that are relevant to applications in the geospatial domain. The relationship that a collection is *extension-of* a universal is, in the context of this research, defined by the group entity Gr_C in Def. 4.5. An example of the *partition-of* relationship between a collection (group) and an individual (object) is that, a map space (individual) is divided by a trans-hydro network [34, 199], which results in a collection of spatial units that partitions the map space. It is required that the partition units are mutually exclusive and jointly exhaustive with respect to the individual being partitioned. Other spatial partitioning features include administrative units, spatial tessellations (e.g. grid- or hexagon-based), Delaunay triangulations and Voronoi regions [1], which are particularly useful in automated generalization and evaluation to divide map spaces into reasonable and computationally tractable units for further processing.

4.1.3 *Spatial relationships between entities*

The above section exemplifies ontological relationships between fundamental categories at an ontological level. This section focuses on a set of relationships between individuals (i.e. spatial entities) that is particularly useful for automated evaluation of generalized datasets. As discussed in previous chapters, characteristics and relationships of (or between) objects, and patterns and structures of groups should be evaluated against the map specifications. This section divides a variety of such relationships and properties into the ones for entities within the same scale (intra-scale relationships) and the ones for entities at different scales (inter-scale relationships).

Intra-scale relationships

Inter-scale relationship is also referred to as *horizontal* relationship [34]. All horizental relationships between individuals in the geospatial domain, $\mathbb{D}$, can be described in terms of qualitative and quantitative relationships. The relationships in topological space are usually qualitative, whereas the ones in metric space are usually quantitative in nature. Qualitative relationships are usually described as predicates or mathematic relations:

$$R(x_1, \ldots, x_n) \subseteq \mathbb{D}_1 \times \ldots \times \mathbb{D}_n$$

where $x \in \mathbb{D}$ is a spatial entity; n is the number of entities involved in a relation, called the relation's *arity*. A truth value can be assigned to a relation, indicating whether some relationship (or property) holds for the involved entities $x_1, \ldots, x_n$. An example of *binary* relation is topological relationships: {*Disjoint, Touch, In, Contain, Overlap, Cross, Equal*}. When we say that a and b are disjoint, we assert $\mathit{Disjoint}(a, b)$. The semantics of topological relationships between point, linear and areal geometries is well-defined in the 9-intersection model [62, 63, 64]; several extensions are also available [e.g. 54].

Some quantitative relationships can be qualified, such as absolute size: {*huge, big, small, tiny*}, relative size: {*bigger, equal, smaller*} and distance: {*near, far*}. The former relationship is a *unary* relation (also called a property), e.g., $\mathit{Small}(x_1)$, and the latter two are *binary* ones, e.g., $\mathit{Bigger}(x_2, x_1)$ and $\mathit{Near}(x_2, x_1)$.

Nevertheless, the qualitative form of relationships is not very useful in describing the change of properties of spatial entities over scale transitions, as is usually the case in the evaluation of generalized data. In addition, the qualification process can be cognitively plausible and context dependent [see e.g. 227]. As a result, I choose to formalize spatial relationships in functions instead of discretizing values on a ratio scale to values on an ordinal or nominal scale. The functional specification of spatial relationships are given below:

$$\begin{aligned} f \quad &: \quad \mathbb{D}^n \to \mathbb{R}^n \qquad \text{(Quantitative)} \\ &: \quad \mathbb{D}^n \to \{\text{true, false}\} \qquad \text{(Qualitative)} \\ (x_1, \ldots, x_n) &\mapsto f(x_1, \ldots, x_n) \end{aligned} \tag{4.7}$$

where $\mathbb{D}^n$ is the domain of function f which takes $x_1, \ldots, x_n$ as its input; $\mathbb{R}^n$ is the codomain (or output space) of f. Although $\mathbb{R}$, or more precisely

$\mathbb{R}^+$, is in most cases sufficient (as spatial relationships mostly result in a single positive value), $\mathbb{R}^n$ is formulated to allow for a larger degree of flexibility. Ai et al. [10] for instance presented a Fourier transform based shape descriptor, which yields a series of coefficient as shape parameters. This case can be formalized as a function: $\mathit{FourierShape}(x) = (c_1, \ldots, c_n)$.

This functional view provides a unified approach to the formalization of qualitative and quantitative relationships. Particularly, mathematic relations or predicates used previously to formalize qualitative relationships can be seen as functions with boolean output. Examples of relationships and properties in a functional form are $\mathit{Size}(x_1) = 4$ mm^2, $\mathit{Distance}(x_1, x_2)$ = 3 mm and $\mathit{Contain}(x_2, x_3)$ = true. Functions are also termed properties or measures.

Contextual relationships Contextual relationship is a special inter-scale relationship. I agree with Mustière and Moulin [140] on their distinction of contextual relationships into: (i) being part of a significant group (e.g. alignments), (ii) being located in a particular area, and (iii) being in relation to the 'same-level' entities.

By the contextual relationship in the proposed formal theory, I mean 'being located in a particular area' only. Being part of a significant group can be described by the *member-of* relationship between groups and their members (see Def. 4.3). The third type of relationships is treated as inter-scale relationships (Def. 4.7) such as $\mathit{AdjacentTo}(b_1, b_2)$ and $\mathit{Parallel}(r_1, r_2)$. The concept of 'same-level' is defined by the division of entities into individuals, collections and universals in this research. The contextual relationship that an object is located in a particular context can be defined as follows:

$$\mathit{InContext}(x) \equiv (\exists c)\,(\mathit{Context}(c) \wedge \mathit{In}(x, c)) \tag{4.8}$$

where $\mathit{In}(x, c)$ is a topological relationship between object x and its context c. Spatial context also requires that $c.\mathtt{CL} = \mathit{Context}$ and $c.\mathtt{Geo.type} = \mathtt{areal}$. For example, in describing a building b_1 located in urban areas, one can write $\mathit{InUrban}(b_1)$, meaning that there exists an object c_1 such that both $\mathit{Urban}(c_1)$ (see Def. 4.6) and $\mathit{In}(b_1, c_1)$ hold. Note that the context instance c is omitted from the notation $\mathit{InContext}(x)$, mainly because that no matter which c the entity x is located in, x is in the context.

In applications where information is available to indicate spatial context, e.g., physical planning data provides parcels indicating whether they are of urban, rural, commercial or forest land-use [68], inferences based on Def. 4.8

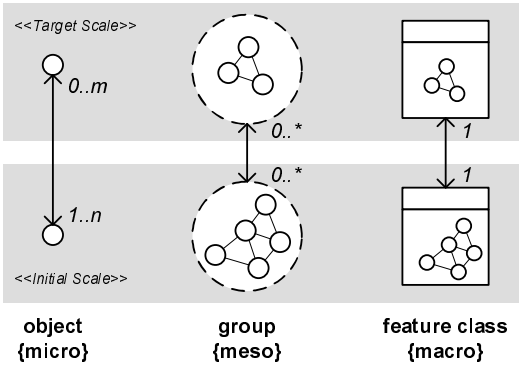


Figure 4.3 Inter-scale relationships between objects, groups of objects and feature classes

can be easily performed by querying over the instances of the context class (e.g. urban polygons). However, in cases where the contextual information is not encoded, e.g., the topographic databases that are currently available, implicit context should firstly be identified before any automated reasoning can be made. This can be done by triggering an operation that classifies the map space into different contextual areas. Steiniger et al. [186] for example proposed supervised classification algorithms that characterize built environments into different subtypes.

Inter-scale relationships

Spatial entities are usually represented at multiple scales for visualization and analysis. The spatial entities of different scales that represent the same real-world phenomena are related (linked) by inter-scale or *vertical* relationships. The inter-scale relationships between entities of different granularities are illustrated in Fig. 4.3. The many-to-many multiplicity of inter-scale relationships is caused by the generalization process. For example, three types of inter-scale relationships can be distinguished between individual objects in terms of multiplicity:

- 1-to-1 or 1-to-0 relationship describes that an initial object has been kept, or removed at the target scale;
- n-to-1 relationship ($n \geq 2$) is caused by merging, aggregating or amalgamating more than one objects to one object;
- n-to-m relationship ($n > m \geq 2$) describes a generalization operation that replace a group of objects by a group of fewer objects that still preserve the spatial distribution of the initial group (i.e. typification).

Note that this research does not consider 0-to-1 relationship between individual objects, since it describes an introduction of new objects into the

target data by an independent updating cycle. Besides, n-to-m relationship between objects can be viewed as a simpler one at a different granularity. If both the n initial objects and the m target objects form significant groups, it can be modeled by 1-to-1 relationship between group entities (Fig. 4.3). Of course the inter-scale relationships between groups can be more complicated, e.g., 0-to-1 and n-to-1 relationships between groups are also possible (see Section 6.3). For the evaluation against feature class level constraints, corresponding relationship between classes of objects (whole datasets) should be modeled. The only type of inter-scale relationship between feature classes is 1-to-1 correspondence (Fig. 4.3).

For simplicity, many-to-many and many-to-one relationships are decomposed into a set of one-to-one (binary) relationships. The same strategy has been applied by Bobzien et al. [34] in their modeling of horizontal, vertical and updating relationships for MRDBs. It is called corresponding relationship and defined as a primitive binary relationship:

$$\begin{aligned} Corr\left(E_i^{Sp}, E_j^{Sq}\right) \quad &: \quad \mathbb{D}^2 \rightarrow \{\text{true, false}\} \qquad \text{(Certain)} \\ &: \quad \mathbb{D}^2 \rightarrow [0,1] \qquad\quad \text{(Uncertain)} \end{aligned} \tag{4.9}$$

The corresponding relationships that are maintained by the generalization process are regarded as links with certainty; whereas the relationship created by an automated matching process are thought of as links with uncertainty, and a value between 0 and 1 should be assigned to the relationship indicating the probability of the matching. This uncertainty value could be used later to indicate the confidence of a evaluation result (see Sections 6.3.3 and 6.4.2).

Summary of spatial relationships

To summarize, there are several types of relationships and properties that is relevant to a meaningful framework for automated evaluation (see Chapter 3). Their abstract syntax are given in Table 4.2. The notation of entities (Def. 4.1) with scales are used in the abstract syntax, and which can be substituted by objects (Def. 4.2), groups (Def. 4.3) or feature class groups (Def. 4.5) with scales in different contexts.

4.1.4 *Meso-groups as examples*

Map generalization usually emphasizes groups of features that have (visually) significant arrangements, patterns and structures, from which underlying geographic processes may be implied (see Section 2.1). In this

Type of relationship		Syntax	Example
Intra-scale	Entity-level	$Exist\left(E_i^{Sp}\right)$	$Exist\,(b_1) = \text{false}$
	Property-level	$Prop\left(E_i^{Sp}\right)$	$Orientation\,(b_1) = 45°$
	Relation-level	$Rel\left(E_i^{Sp}, E_j^{Sp}\right)$	$Touch\,(b_1, b_1) = \text{true}$
	Contextual	$InContext\left(E_i^{Sp}\right)$	$InRural\,(b_1) = \text{false}$
Inter-scale		$Corr\left(E_i^{Sp}, E_j^{Sq}\right)$	$Corr\,(r_1, r_2) = 0.85$

Table 4.2 Syntax of different types of relationships

section, examples are given on meso-level groups, namely networks and alignments, using the formalism described in Sections 4.1.1–4.1.3.

Networks

Common geographic networks include transportation and hydrology networks. Network features are characterized by elementary properties such as connectivity of linear features (e.g. segments) and collective ones such as spatial distributions and patterns of the networks (see Section 2.1). A network feature is given by a group entity (Def. 4.3) with a connectivity relationship:

$$Gr_{net} = \langle \text{ID},\ OC = \{o_1, \ldots, o_n\},\ Connectitiy\,(OC)\,,\ Geo \rangle \tag{4.10}$$

where it is required that $(\forall o_i)((o_i \in OC) \Rightarrow (o_i.Geo.type = linear))$; the symbol '$\Rightarrow$' is logic implication and read as 'implies'. Geo is the representative geometry of the network, which can be a convex hull indicating its extent, or a set of polygons implementing the concept of influential regions of linear features of the network (e.g. catchments of rivers of a drainage system). In both cases, $Gr_{net}.Geo.type = areal$.

By introducing the concept of path, p, it is now possible to define the relationship connectivity as:

$$\begin{aligned} Connectitiy\,(OC) \equiv & \left(\forall o_i, o_j\right)\left(\left(o_i, o_j \in OC\right) \Leftrightarrow Touch\left(o_i, o_j\right) \vee \right. \\ & \left. (\exists p)\left(\,(p \subset OC) \wedge Touch\left(o_i, p\right) \wedge Touch\left(o_j, p\right)\right)\right) \end{aligned}$$

where $p = \{o_1, \ldots, o_m\}$, $(m < n)$ is a sequence of connected spatial features such that $(\forall o_i, o_{i+1})((o_i, o_{i+1} \in p) \Rightarrow Touch(o_i, o_{i+1}))$. The above definition of connectivity reads as: we say that any two objects are elements of a spatially connected set OC, if and only if either the two objects touch

each other, or the set OC contains a path p that touches both objects. The semantics of *Touch* is well-defined with point set topology [e.g. 54, 62]. This definition of connectivity can also be applied to express connected polygons such as 'aggregated building' in [160] and 'terraced house' in [113]. For geographic networks in particular, it is required that all objects are linear features ($o_i.Geo.type = linear$).

More generally, a binary relationship that hold between connected objects can be defined with the help of the concept of path p:

$$Connected(o_i, o_j) \equiv (\exists p)\left(Touch(o_i, p) \wedge Touch(o_j, p)\right) \vee Touch(o_i, o_j) \tag{4.11}$$

This relationship is useful in specifying the constraint that intends to preserve the connectivity of network features during generalization.

Alignments

An alignment is a linear pattern which is composed of individual objects (e.g. buildings, ponds, islands, etc.) that arrange themselves on a virtual (curve) line. Such linear patterns are of significant value in the generalization domain (see Section 2.1). An alignment feature is given by a group entity (Def. 4.3) with a relationship called *Aligned*:

$$Gr_{align} = \langle ID, \{o_1, \ldots, o_n\}, Aligned(o_1, \ldots, o_2), Geo \rangle \tag{4.12}$$

where Geo is the representative geometry and usually the skeleton of the alignment [239]. This skeleton can be used to indicate the virtual (curve) line to which the group members align, so it is required that $Geo.type = linear$. One should also note that the skeleton is calculated after the alignment has been recognized rather than exists *a priori*. Typically, an alignment consists of at least three member objects so that the mean and standard deviation of many properties can be measured for its members. As a result, it is required that $|Gr_{align}.OC| \geq 3$.

The spatial relationship, *Aligned*, that bands the group members together is based on a general notion of 'consistent' and can be defined recursively:

$$\begin{aligned} Aligned(o_1, o_2) &= Consistent(\{o_1\}, o_2) \\ Aligned(o_1, o_2, o_3) &= Consistent(\{\{o_1\}, o_2\}, o_3) \\ &\ldots \\ Aligned(o_1, o_2, \ldots, o_n) &= Consistent(\{\ldots\{\{o_1\}, o_2\}, \ldots\}, o_n) \end{aligned}$$

where *Consistent*({_}, _) is a relationship (or boolean function) that takes a set as its first argument and an object as its second argument. This definition can be interpreted as follows: an alignment can be recursively expanded by taking another object in if the object is consistent with the existing group; it starts with identifying a pair of aligned objects as a basic group, and grows until there is no other objects that is consistent with the group. This definition is generic enough to cover alignments of point-like, linear and areal features. For example, several algorithms are designed in this research to recognize clusters and alignments of areal objects (e.g. buildings) in Section 5.4, where the notion of 'consistent' combines the relationships of proximity, path angle, align angle, size, orientation and shape of the areal feature. This research will not fully explore the semantics of 'consistent' as it is out of the scope.

Also in the recursive definition, sets are expressed as nested sets in which the order of objects in an alignment is visiable. Formally, an alignment can be viewed as an ordered list of objects (or a sequence) where the order of the composing objects is important. For example, (o_1, o_2, o_3) is a sequence of objects that differs from (o_1, o_3, o_2). Nested sets better capture this information than do ordinary sets as ordering is imposed in the structure of nested sets, e.g., $\{\{\{o_1\}, o_2\}, o_3\} \neq \{\{\{o_1\}, o_3\}, o_2\}$. Note that, although alignments of spatial objects are usually a special type of clusters [234, 238], the ordering concept does not apply to cluster members.

Major properties of alignments include spatial distribution and orientation. Alignments, or clusters in general, are regular spatial distributions, so I use the notion of homogeneity to quantify the degree of regularity of alignments or clusters. It can be formally expressed as $\mathit{Homo}(al_i) : \mathbb{D} \rightarrow [0, 1]$. Note that a group is an entity such that $al_i \in \mathbb{D}$. Similarly, $\mathit{Orientation}(al_i) : \mathbb{D} \rightarrow \mathbb{R}^+$ expresses the orientation of an alignment. Parallel relationship between alignments, $\mathit{Parallel}(al_i, al_j) : \mathbb{D}^2 \rightarrow [0, 1]$, is also useful if two alignments (especially curvilinear ones) are to be compared in terms of orientation. Mathematical definitions of these functions are given in Sections 5.3.2 and 5.4.5.

4.2 Characterization and evaluation functions

Automated evaluation requires to characterize properties of and relationships between spatial entities (characterization functions), and to evaluate their target states with respect to the initial states (evaluation functions). The role of characterization and evaluation functions and their formalism

were described in [19, 21, 120]. Characterization functions can be formalized using the intra-scale relationship defined in Def. 4.7 and can be divided into three levels: entity-level, property-level and relation-level (shown in Table 4.2).

Based on the results of characterization functions and conditions specified in cartographic constraints, evaluation usually requires to compare initial and target states (properties or relationships) of objects (Def. 4.13a), to normalize the differences into violations or satisfactions of constraints (Def. 4.13b), to qualify the results into information that is meaningful to end users (Def. 4.13c), and to aggregate individual assessments into a global ones (Def. 4.13d).

$$\mathit{Comp} \quad : \quad \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R} \tag{4.13a}$$

$$\mathit{Norm} \quad : \quad \mathbb{R} \rightarrow [0,1] \tag{4.13b}$$

$$\mathit{Qual} \quad : \quad [0,1] \rightarrow \{\text{quality descriptions}\} \tag{4.13c}$$

$$\mathit{Aggr} \quad : \quad [0,1]^k \rightarrow [0,1] \tag{4.13d}$$

To explain, *Comp* is used when the constraints need to preserve a given property, and it can be defined differently depending on the property to be compared, e.g., $\mathit{Size}(o_1)/\mathit{Size}(o_2)$ or $\mathit{Orientation}(o_1) - \mathit{Orientation}(o_2)$. The normalization function takes the result from *Comp*, or characterization functions for readability constraints, as input, and results in a value that can be interpreted as the degree of violation (CV) or satisfaction (CS). Note that for cases where functions result in boolean output (false = 0 and true = 1), normalization is not necessary.

The evaluation results (CV or CS) of individual constraints can be viewed as individual assessments, which can be aggregated for a whole dataset. The parameter k in Def. 4.13d means the number of entities (e.g. objects or groups) in a dataset. For feature class level constraints, there is only one extension of a particular class in a dataset ($k = 1$). Aggregation function usually takes the form of weighted average, with weights indicating the relative importance of the entities in a dataset (see e.g. Section 6.4.2). The aggregation goes further when someone combines CV or CS of individual constraints summarized for a dataset to calculate the overall quality of the map. In this case the weights are set according the importance of constraints. Finally, the individual and global assessments can be qualified using *Qual* into quality descriptions, e.g., { good, nearly good, nearly bad, bad }.

4.3 CONSTRAINTS AS LOGICAL EXPRESSIONS

In expressing cartographic constraints, several side notes should be considered. First, the domain of discourse $\mathbb{D}$ is subdivided into sub-domains $\mathbb{D}^{S1}, \ldots, \mathbb{D}^{Sn}$, distinguishing between entities at different scales. So entities such as o^{S1} and Gr^{S2} are instances or collections of respective sub-domains. Second, aliases are frequently used to simplify the expressions, the meaning of which is given in specific contexts (e.g. $align_1$ is a Gr_{align}). Third, the conditions to be respected by constraints are formalized as conditional expressions, where thresholds are left as user-defined parameters μ, e.g., $Size(o_1) \geq \mu$. For simplicity, a boolean condition is written shortly as e.g. '$Touch(a, b)$' without a '= true' clause. In both cases, the conditional expressions fit into the framework of predicate logic. In the following, I give examples of formalized constraints which are drawn from NMA specifications for the generalization from 1:10k to 1:50k [e.g. 192]. The choice of the constraints considers the order of expressive complexity and the requirement for balanced constraints (see Section 3.4), which does not necessarily indicates the importance of the constraints.

To begin with, a minimum dimension constraint (C1) saying that area of any target polygon should be larger than a certain threshold can be formalized as:

$$(C1) \quad \left(\forall o_i^T\right)\left(\left(o_i^T.Geo.type = areal\right) \Rightarrow Size\left(o_i^T\right) \geq \mu\right)$$

Note in the above expression that, o_i^T, with T meaning target scale, is used to simplify the expression $(o_i)(o_i \in \mathbb{D}^{50k})$ in this context. Likewise, o_i^I is used to indicate that it is an entity of initial scale (1:10k). The same is used throughout the following expressions. A second constraint (C2) is a Dutch map requirement saying that important buildings in initial data should be kept (i.e. entity-level preservation):

$$(C2) \quad \left(\forall b_i^I\right)\left(ImportantBuilding\left(b_i^I\right) \Rightarrow \left(\exists b_i^T\right)\left(Corr\left(b_i^I, b_i^T\right)\right)\right)$$

where b_i is an alias of an instance of the building class in this case; the type predicate (Def. 4.6) restricts the range of the quantifier to important buildings. C2 explicitly states that for every important building in the initial data, there exists at least one building in the target data that correspond to it (Def. 4.9). If one intends to express the idea that there is one and only one target object which corresponds to the initial object, the term after the implication symbol in C2 can be rewritten as:

$$\left(\exists o_i^T\right)\left(Corr\left(o_i^I, o_i^T\right) \wedge \left(\forall o_j^T\right)\left(Corr\left(o_i^I, o_j^T\right) \Rightarrow \left(o_i^T = o_j^T\right)\right)\right)$$

The third constraint (C3) is a preservation constraint with added limits (scopes) to the entity class to be considered. It requires that target shape should remain concave if the initial polygons are of high concavity:

$$\text{(C3)} \quad \left(\forall a_i^I\right)\left(\left(\mathit{Concavity}\left(a_i^I\right) \geq \mu\%\right) \wedge \left(\exists a_i^T\right)\left(\mathit{Corr}\left(o_i^I, o_i^T\right)\right.\right. \\ \left.\left. \Rightarrow \mathit{Similar}\left(\mathit{Concavity}\left(a_i^I\right), \mathit{Concavity}\left(a_i^T\right)\right)\right)\right)$$

where we use a_i as an alias of the expression $(o_i)(o_i.\mathtt{Geo.type} = \mathtt{areal})$. Note that, the degree of concavity *Concavity* has to be defined at the implementation level, and the concept of high concavity can be specified by users (e.g. $\mu = 95\%$). An abstract comparison function *Similar* is used here since the concrete form of the function is unknown at this moment, which can be replaced if the concrete function is provided. Note that the scope that limits the polygons in the initial data states that C3 only considers the initial objects (i) that are highly concave in shape and (ii) that were kept in the target data. This implies that the objects that were deleted during the generalization are not considered by C3. The next example shows a constraint requiring that proximate and roughly parallel features (line-line or polygon-polygon) should become topologically adjacent (C4):

$$\text{(C4)} \quad \left(\forall o_i^I, o_j^I\right)\left(\left(\mathit{Distance}\left(o_i^I, o_j^I\right) < \mu_1\right) \wedge \left(\mathit{Parallel}\left(o_i^I, o_j^I\right) \geq \mu_2\%\right) \wedge \right. \\ \left. \left(\exists o_i^T, o_j^T\right)\left(\mathit{Corr}\left(o_i^I, o_i^T\right) \wedge \mathit{Corr}\left(o_j^I, o_j^T\right) \Rightarrow \mathit{Adjacent}\left(o_i^T, o_j^T\right)\right)\right)$$

where μ_1, distance, and μ_2, degree of parallelism, are given by users to define proximity and parallelism between objects. Similar to C3, C4 only considers the initial object pairs where the two objects are close and parallel to each other and where both objects were kept in the target data. This is an example where the topological relationship between objects should change. Next, a constraint (C5) that preserves topological connectivity between objects is exemplified on top of the relationship *Connected* (Def. 4.11):

$$\text{(C5)} \quad \left(\forall o_i^I, o_j^I\right)\left(\mathit{Connected}\left(o_i^I, o_j^I\right) \wedge \right. \\ \left. \left(\exists o_i^T, o_j^T\right)\left(\mathit{Corr}\left(o_i^I, o_i^T\right) \wedge \mathit{Corr}\left(o_j^I, o_j^T\right) \Rightarrow \mathit{Connected}\left(o_i^T, o_j^T\right)\right)\right)$$

where o can be either linear or areal features. C5 can be used to check whether the connectivity has been kept between any two objects or for network features. Consider the following variation of C5:

$$\text{(C5')} \quad \left(\forall net_i^I\right)\left(\exists net_i^T\right) \mathit{Corr}\left(net_i^I, net_i^T\right)$$

where net is an alias of $\mathtt{Gr}_{net}$. One may argue that the existence of a corresponding network in the target data automatically ensures the connectivity of the initial network being kept, as it appears that connectivity is a built-in

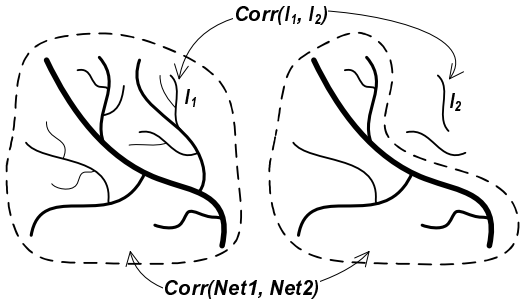


Figure 4.4 Connectivity between river branches in a network is violated (networks as groups are highlighted by dashed lines): initial network feature (left) and generalized feature (right)

property of a network feature (see Def. 4.10). This is however not true as can be shown in Fig. 4.4. In this illustration, the initial network (the group on the left) was kept after generalization (the group on the right), but some river branches such as l_2 become disconnected from the initial network. Such a violation to connectivity can be identified by C5 but not C5'.

Finally, some examples is given about the constraints on alignments (collections). The first one (C6) requires that alignment should be kept (an entity-level preservation); the second one (C7) requires that for the target alignment its orientation should be similar to the initial one; the third one (C8) requires that the spatial distribution (regularity) of the members of the target alignment should keep or even enhance the regularity of the members of the initial alignment.

$$\text{(C6)} \quad (\forall al_i)\left(al_i \subset \mathbb{D}^I\right) \Rightarrow (\exists al_j)\left(al_j \subset \mathbb{D}^T\right) \wedge Corr\left(al_i, al_j\right)$$

$$\text{(C7)} \quad \left(\forall al_i^I\right)\left(\exists al_i^T\right)\left(Corr\left(al_i^I, al_i^T\right)\right)$$
$$\Rightarrow \left(\left|Orientation\left(al_i^I\right) - Orientation\left(al_i^T\right)\right| \leq \theta\right)$$

$$\text{(C8)} \quad \left(\forall al_i^I\right)\left(\exists al_i^T\right)\left(Corr\left(al_i^I, al_i^T\right)\right) \Rightarrow \left(Homo\left(al_i^T\right) \geq Homo\left(al_i^I\right)\right)$$

4.4 Discussion and reflections

4.4.1 *Expressiveness of the formal theory*

The formal theory described in this chapter actually formalizes the syntactic structure of cartographic constraint described in Section 3.2.4: entity class, scope modifier, constrained properties, and conditions. This means the formal theory is able to precisely express all types of constraints that can be combined from these syntactic building blocks. The expressive power comes especially from the ability to precisely define the scope modifier that

limits the entity class(es) to be considered by a constraint. For example a constraint 'the road leading to a building in a peninsula should be kept' can be formalized as follows:

$$\begin{aligned}&\left(\forall r_i^I, b_i^I\right)\left(\mathit{LeadTo}\left(r_i^I, b_i^I\right) \wedge \mathit{InPeninsula}\left(b_i^I\right)\right)\\ &\quad\Rightarrow \left(\exists r_i^T, b_i^T\right)\left(\mathit{Corr}\left(r_i^I, r_i^T\right) \wedge \mathit{Corr}\left(b_i^I, b_i^T\right)\right)\end{aligned}$$

In this example, the range of quantification is specified through scope modifiers, i.e., the relationship *LeadTo* and the contextual relationship *InPeninsula* (see Def. 4.8). The example constraint states that for all initial roads and buildings that satisfy that buildings are located in peninsulas and roads lead to those buildings, there exists at least a pair of target objects that corresponds to each pair of initial roads and buildings.

In many places, many-sorted logic is adopted to simplify (one-sorted) predicate logic. This is done by adding domain predicates to divide the domain of discourse into sub-domains (types). To strictly follow one of them is not essential, as many-sorted logic provides exactly the same expressive power as (one-sorted) logic, and there is a translation between the two [66]. For example, the expression that all target buildings should be large enough to be visible: $(\forall b_i^T)(\mathit{Size}(b_i) \geq \mu)$ can be translated to a one-sorted expression: $(\forall o_i)(o_i \in \mathbb{D}^T \wedge \mathit{Building}(o_i) \Rightarrow (\mathit{Size}(o_i) \geq \mu))$. In this respect, many-sorted expressions can be regarded as abbreviations or aliases for longer (one-sorted) expressions. Both many-sorted (e.g. C5′ and C7) and one-sorted expressions (e.g. C6) are used in the presented formalism.

4.4.2 *Semantics of spatial relationships*

As mentioned in this chapter, the semantics of topological relationships are extensively studied and well-defined along with topological space (point set topology). As to higher-level relationships such as contextual relationships and those imposed by significant groups, their semantics can be difficult to define. For instance, although it is possible to precisely define the connectivity of network features (see Def. 4.11) on top of well-defined topological relationship, the structures and patterns of networks may only be defined statistically (e.g. via statistical analysis of the major flow directions of the river branches).

Another example is alignments of spatial objects. In the semantic aspect, Billen and Clementini [28] formally defined the semantics of collinearity for three and more regions. In their approach, collinearity is based on the relationships in projective space, which is extended to define collinearity among

three regions by viewing regions as sets of points. The authors claimed that by joining up step-wise local collinearity, a global collinear or curvilinear arrangement may occur. This claim sounds interesting, as collinearity and curvilinearity is also important characters of alignments. However, their definition of collinearity does not lead to significant alignments, because in addition to collinearity, the alignment relationship has to combine other metrical relationships such as distance, orientation, size, shape, density, etc (see Section 5.4). The definition of collinearity in [28] that arises purely from projective properties is not enough to describe alignments that are geographically meaningful. Besides, their claim about the derivation of curvilinearity is not stable and hence cannot be used as a definition of curvilinear alignment.

It has to be mentioned that research on semantics of spatial relationships originates from Artificial Intelligence (AI) domain, and its purpose is to enable qualitative spatial reasoning, in which the determination of spatial relationships between data objects is not a major concern. This thesis focuses on the one hand on the formalization of map specifications (constraints), and on the other hand on measuring and detecting relationships to enrich the databases for automated evaluation. Hence the semantic modeling of spatial relationships for qualitative spatial reasoning is not necessary. This explains why in automated generalization, quantitative relationships are the most frequently used and defined in the form of measures or algorithms, i.e., the procedure that determines the relationship between objects. Defining the semantics of quantitative relationships can be very complicated (if not impossible), since the relationships can only be fully described in metric space.

4.4.3 *Possibilities for automated evaluation*

A formal approach to automated evaluation allows to engage the reasoning facilities in a logical system. Ideally, this requires to explicitly define all entities, properties and relationships at the data level (e.g. in a relational database) to ensure a fully automated reasoning. However, due to the amount of required information and the purpose-dependent nature of generalization, it is impossible to store all the information and some information has to be calculated on-demand. Current research on automated generalization focuses partly on developing various methods and algorithms to enrich existing (raw) data with extra information and knowledge. From the perspective of a formal methodology, data enrichment operations should be semantically well-defined to be interoperable with the logical unit (i.e. automated reasoning). For example, when the logical unit identifies that an

expression requires information that is not defined, it automatically triggers 'appropriate' operations (algorithms) to calculate the required information on-the-fly, and retrieves the result to proceed the reasoning. The 'appropriateness' of the choice of operations should rely on a semantic matching between the logical expression and potential operations, rather than on a trial-and-error approach. The integration of the operational aspects of a system into an automated reasoning framework requires more researches on the semantic aspects of spatial operations and algorithms in the future. In this respect, Haarslev and Möller [80], Haarslev et al. [81, 82] for instance have proposed to integrate topological relationship with a logical framework for automated qualitative spatial reasoning.

The disadvantages of using boolean logic for automated evaluation is also obvious. First, in the inference process, a constraint is evaluated to be either true or false in terms of constraint violation (or satisfaction). It cannot handle intermediate states that may also be considered as acceptable solutions. For example, some constraints may be deliberately relaxed to meet more important constraints in specific contexts. Second, the degree of flexibility that is usually suggested in a manual evaluation as discussed in Section 3.3.2 can be very difficult to implement in a logical framework. In other words, evaluation based on logical reasoning is not able to handle the evaluation for summarizing quality descriptions and for comparing different generalization solutions. As a result, one has to resort to the evaluation in a non-logical framework. That is, the generalized data are evaluated by calculating individual assessments, i.e., degrees of constraint violation (or satisfaction). Subsequently, the individual assessments can be aggregated using the cost function discussed in Section 3.3.3. This way, the flexibility used in manual generalization and visual assessment can be allowed for in an automated approach.

Given the above deficiencies of the logical framework, the proposed formal theory still provides a powerful way to precisely define cartographic constraints. In addition, automated evaluation in a logical framework is still effective in identifying cartographic errors and ensuring the integrity of databases [194]. Note that in both cases (logical and non-logical), the concepts, properties, and relationships specified in cartographic constraints have to be made explicitly available to support automated evaluation.

4.4.4 *Implementation of the theory*

This chapter focuses on the modeling and formalization principles. Hence the implementation of the theory (i.e. data enrichment, data matching, and

selection of formal languages with reasoning capabilities) is a secondary issue. In the following chapters, I will concentrate on data enrichment (Chapter 5) and matching techniques for the evaluation (Chapter 6). The selection and testing of formal language will not be addressed in this thesis. Nevertheless, to the best of my knowledge, the Object Constraint Language (OCL) [216] has the potential to implement the formal theory presented in this chapter and to formalize map specifications together with a UML model. Research in the geospatial domain has demonstrated the use of OCL and UML in modeling topological rules in spatial databases [25] and specifications for multiple representation databases and generalization [61, 69, 71, 194].

5 Design of data enrichment

5.1 Introduction

Data enrichment can be generally defined as a process that 'adds various types of information to an existing dataset to make it more valuable for a properly defined application' [49]. In automated generalization domain, data enrichment is a prerequisite to enable automated processes.

For the purpose of automated evaluation of generalization output, several types of information should be enriched from spatial data: (i) implicit geometries (usually parts or aggregations of existing geometries) which is always labeled by some implicit semantics, (ii) implicit semantics of existing objects (e.g. detached houses inferred from building objects), (iii) groups (patterns and networks) that are not explicitly defined and stored, and (iv) implicit properties of, and relationships between, existing or implicit entities. Besides, the corresponding relationship is also one type of information that should be enriched. Unlike the implicit entities, properties and relationships, the corresponding relationship is a logical relation in (multi-scale) database domain and does not have a correspondence in reality.

This chapter develops and refines data enrichment techniques for the purpose of automated evaluation, with a major focus on building class[1]. The presented data enrichment includes individual level information (e.g. size, distance, shape, and orientation) in Section 5.2, group level entities (building clusters and alignments) in Sections 5.3 and 5.4, and several attempts at measuring spatial distributions of feature classes (Sections 5.5 and 5.6). The choice of these topics is justified in Section 1.2 and the requirement for balanced map specifications (Section 3.4).

[1] The evaluation process demonstrated in this thesis uses constraints defined for buildings and occasionally road features as a proof-of-concept. Hence data enrichment also focuses mainly on these feature classes. For a more detailed explanation, the readers are referred to Section 1.3.1

OWN PUBLICATIONS

This chapter is partly based on the following publications:

- X. Zhang, T. Ai, and J. Stoter. The evaluation of spatial distribution density in map generalization. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Congress 2008)*, volume XXXVII. Part B2., pages 181–188, Beijing, 2008.
- X. Zhang, T. Ai, and J. E. Stoter. A voronoi-like model of spatial autocorrelation for characterizing spatial patterns in vector data. In *Proceedings of the 2009 Sixth International Symposium on Voronoi Diagrams (ISVD09)*, pages 118–126, Washington, DC, USA, 2009. IEEE Computer Society.
- X. Zhang, T. Ai, and J. Stoter. Characterization and detection of building patterns in cartographic data: Two algorithms. In *Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science (SDH2010)*, volume XXXVIII, Part 2, pages 261–266, 2010 (paper selected as Chapter 8 in *Advances in Spatial Data Handling and GIS*, Lecture Notes in Geoinformation and Cartography, to appear in 2012).
- X. Zhang, T. Ai, J. Stoter, M.-J. Kraak, and M. Molenaar. Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. *GeoInformatica*, page 33 (online first). doi: 10.1007/s10707-011-0146-3.

5.2 DATA ENRICHMENT OVERVIEW

An overview of the properties and relationships that should be enriched for the purpose of automated evaluation is depicted in Table 5.1. Among others, we describe several measures of properties and relationships that are used extensively in implementing the concepts in this research. These measures include distance (Section 5.2.1), orientation (Section 5.2.2), shape (Section 5.2.3), and proximity (Section 5.2.4), which are the building blocks in the data enrichment and matching of spatial objects.

5.2.1 *Distance*

Distance between spatial entities of different geometry types can be measure based on Euclidean distance between two points of the entities. The choice

Properties or relationships	Example measures	Implementations
Size, distance	Euclidian metrics	EE or SF
Position	center of gravity (centroid)	EE or SF
Orientation	longest edge, rotated minimum bounding box (RMBB), wall statistical weighting, etc.	[59]
General shape	Fourier shape descriptor and similarity measure	[10]
Specific shape	squareness, elongation, shape index, fractal dimension, sinuosity, etc.	[2]
Shape of part of geometry	bends/corners detection, shape of bends (e.g. depth, elongation), etc.	[206, 207]
	bend hierarchy	[8]
Topological relationships	touch, disjoint, equal, etc.	EE or SF
Neighborhood (proximity)	Delaunay triangulation based neighborhood	[6, 100]
Parallelism	parallelism between arbitrary shapes	
Spatial distribution	global spatial distribution: *random*, *regular*, and *clustered*	[36, 95]
	density based similarity measure	[235]

Table 5.1 Properties and relationships that can be enriched from existing data (EE – Explicitly Encoded; SF – Spatial database Functionalities)

of point(s) depends on what you want to measure, e.g., an overall distance or a minimum distance between objects. The simplest way to measure distance between centroid points of objects. This is suitable for measuring the positional similarity for points and polygons with compact and simple shapes. The similarity of polygons can also be measured by their (relative) overlap, e.g., the contrast model proposed by Tversky [202]. For linear features, *Hausdorff* distance and *Fréchet* distance are commonly used [12, 139] to measure the distance and similarity. Fréchet distance, in particular, is more appropriate to compare very sinuous lines such as coastlines [12, 125].

Readability constraints usually require that minimum distance between objects should be ensured. Nearest distance is widely used to measure the minimum distance between two objects. Nearest distance between A and B can be defined as: $\exists p \in A, q \in B, \mathit{NearestDis}(A, B) = d(p, q)$ such that

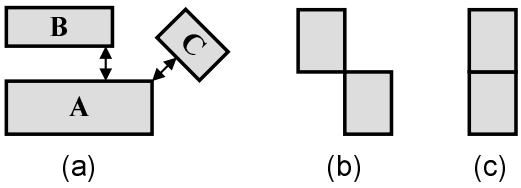


Figure 5.1 Nearest distance and its problems in cognition

$\forall p_i \in A, q_i \in B, d(p, q) \leq d(p_i, q_i)$. At an operational level, Delaunay triangulation (Section A.1.1) can be used to assist the computation of Euclidian distance and especially the nearest distance between any proximate geometries. The idea and advantages are described in detail in Section A.1.2.

Note that, it is commonly the case in topographic data that pairs of objects with the same nearest distance are cognitively different in perceived distances. Examples of this are illustrated in Fig. 5.1. For instance, some people might have a perception that in Fig. 5.1(a), A is closer to B than is to C though $\mathit{NearestDis}(A, B) = \mathit{NearestDis}(A, C)$. Fig. 5.1(b) and (c) show a similar situation, where the two pairs of polygons with a same nearest distance (0 mm) might be viewed different in cognition. Such cases have also implications in the evaluation of cartographic constraints. Studies in the EuroSDR project have identified situations that 'violate' the minimum distance constraints, as in the case of Fig. 5.1(b), which are nevertheless considered to be acceptable by cartographers [42, 192]. This may call for a redefinition of the constraints for specific cases, or it can be addressed by incorporating cognitive aspects into the measure. Ai and Zhang [7] for instance proposed a distance measure based on Delaunay triangulation that takes cognitive and contextual aspects into account. A detailed account for this measure can be found in Section A.1.2.

5.2.2 *Orientation*

Orientation is a fundamental property of spatial entities in generalization. Here we describe measures of orientation for areal (buildings in particular) and linear features.

Orientation of building polygons

The measure of building orientation is based on wall statistical weighting (WSW) described in Duchêne et al. [59]. The resulting orientation is wall direction α in $[0, \pi/2]$. The result also comes with a confidence value (numbers indicated in Fig. 5.2(a)), which is calculated by counting the proportion of length of the edges that orient to $\alpha \pm \sigma$ (a small tolerance). Typical build-

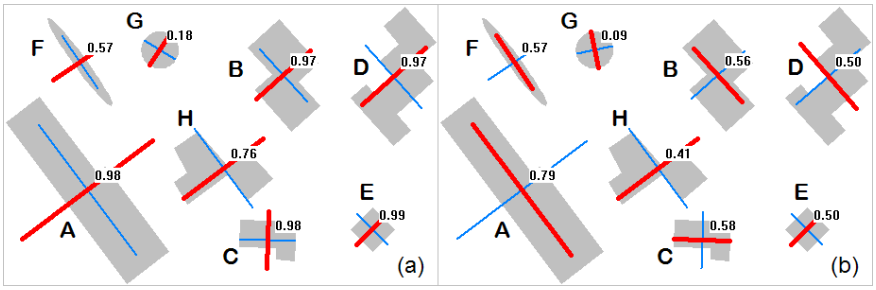


Figure 5.2 Building orientation measures: (a) wall statistical weighting; (b) adaptation in this approach (output orientations are in red bold lines with confidence values numbered upper-right)

ings have two perpendicular wall directions (α and $\alpha + \pi/2$). The walls of direction $\alpha + \pi/2$ also add to the confidence value of the resulting WSW orientation α (e.g. confidence values of A - E in Fig. 5.2(a) approach to 1).

In this paper, we adapted the original WSW, in which we distinguish wall direction α from $\alpha + 90°$. The output orientation is the direction (in $[0, \pi]$) in which lengths of walls accumulate most; in most cases this is the dominant one (i.e. major wall direction) of the two perpendicular directions. After adaptation the output orientations adjust better to their major wall directions (e.g. A, B, C in Fig. 5.2(b)). Confidence value decreases accordingly in our adaption since walls of direction $\alpha + \pi/2$ do not add to walls of α (e.g. A - E, H in Fig. 5.2(b)). The adapted confidence value is now correlated with the degree of elongation (strength of major wall direction). A square (E) with a weak major wall direction has a low confidence (0.5); an oval (F) with a strong major wall direction has a relatively higher confidence (0.57). Note that this adaption is by no means to describe a general orientation, but to choose from among the wall directions the most significant one (in $[0, \pi]$). However, except for round shapes (G) the adapted WSW is sufficient for measuring the similarity of building orientations even in the case of stair-like shapes (D). After generalization, stair-like shapes should remain their wall directions to keep their characteristics, but their general orientations may change.

Orientations in linear features

There are several kinds of orientations useful when characterize linear features for cartographic applications (see Fig. 5.3). An average orientation ($AverageO_L$) of a linear feature can be obtained by a summation of its seg-

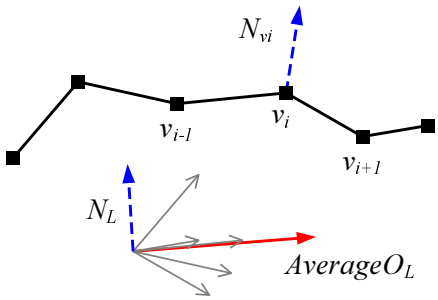


Figure 5.3 Different kinds of orientations in linear features

ment vectors $\overline{v_i v_{i+1}}$, with each vector being weighted by the segment length $|\overline{v_i v_{i+1}}|$. The average normal direction (N_L) of the feature is the direction that is perpendicular to $AverageO_L$. At a local level, the normal direction of any non-end vertex (N_{vi}) in a linear feature can be calculated by the bisector of vectors $\overline{v_{i-1} v_i}$ and $\overline{v_i v_{i+1}}$. This measure is useful in recognizing curvilinear building alignments (see 'CURVILINEAR ALIGNMENTS' in Section 5.4.3).

Practically, topographic datasets contain a large number of linear features that are very sinuous and long. Hence $AverageO_L$ is to a large extent meaningless for describing the general direction of such features. In those cases, one can segment the very long linear features into more homogenous fragments which can be characterized using $AverageO_L$.

For straight line segments and polygons with straight facing edges, the parallelism can be measured by comparing the orientations of the segments or the facing edges. In more generally terms, we say that line a is parallel to b if a has everywhere the same distance to b. Based on this definition, parallelism could be measured between arbitrary shapes such as curves, or one can identify homogenous parts of objects that are parallel to each other. Section A.1.2 describe an idea to measure the (degree of) parallelism between both linear and areal features .

5.2.3 *Shape*

A general shape measure that captures all aspects of shape and maps it into a unique real number proves to be not possible [106]. Practically, researchers work on specific aspects of shape. Examples of the aspects related to cartographic analysis are: complexity, elongation, concavity, squareness, etc.

When it comes to building polygons, these different aspects of shape have

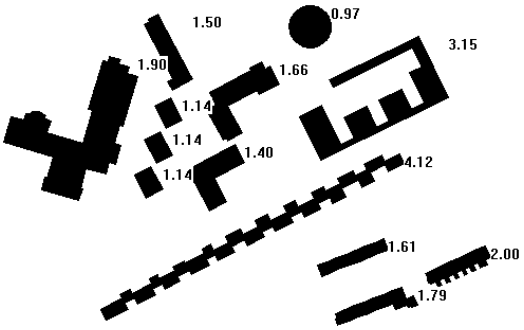


Figure 5.4 Shape index calculated for typical building shapes (values indicated at up right of the shapes)

varying expressive power. For instance, squareness measures the degree to which all corners of a building is right-angled (90° or 270°). Since typical building corners are nearly right-angled, squareness is a less useful measure. Furthermore, since elongation is measured based on the RMBB of a building, it is an indirect measure. Concavity, on the other hand, cannot distinguish the difference between rectangles with different elongations. To summarize, although the above-mentioned measures can be quite useful to describe certain aspects of polygons in general (see e.g. constraint C3 in Section 4.3), they may not be appropriate for comparing the similarity of building shapes.

To compare the similarity of building shapes, we use *shape index* [2, 149] which is able to describe the complexity of building shapes:

$$ShapeIndex(A) = \frac{Perimeter(A)}{2 \times \sqrt{\pi \times Area(A)}} \tag{5.1}$$

where A is a simple polygon. The idea is to compare the perimeter of a polygon to that of a circle with the same area. As a result, the minimum value 1 indicates a circle; more complex shapes have higher values. More specifically, squares have values of approximately 1.12; elongated rectangles have higher values than less elongated ones (e.g. squares); concave and complex polygons also have higher values than less concave and less complex ones. Examples are shown in Fig. 5.4.

5.2.4 *Proximity relationship*

Proximity is a spatial relationship that is not explicitly modeled in common spatial data models. In our definition, proximity describe a relationship between two objects that are neighbors, meaning that they can get access to each other without being blocked by intermediate objects. This definition can be also true for objects at a relative large distance from each other.

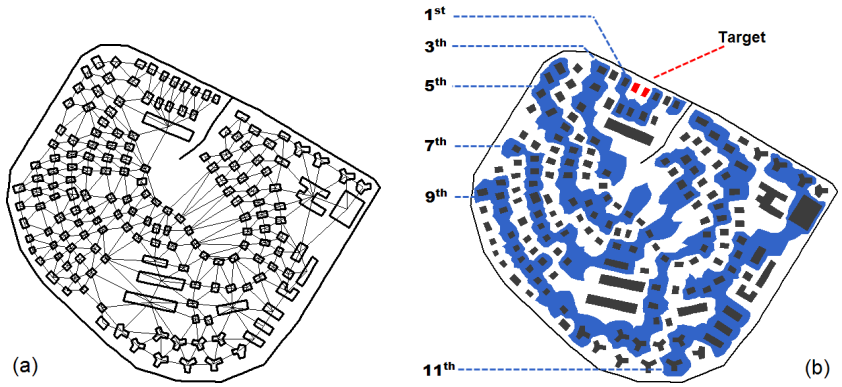


Figure 5.5 Proximity graph (a) and orders of proximity with respect to the target (b) (Data source: Shenzhen, China)

Sometimes we call it as 'AdjacentTo' which differs from a topological 'AdjacentTo' relationship, though the latter can also be interpreted as a proximity relationship. Delaunay triangulation (DT) is used to model the natural relationship of proximity or neighborhood [see e.g. 13, 76, 100]. In this thesis work, DT is used to model proximity relationship and to support other kinds of data enrichment tasks. Here we use buildings and road centerlines to illustrate the idea of proximity. In Fig. 5.5(a), a proximity graph is derived from the DT built on buildings and road lines, where only the proximity relationship between buildings are shown. Note that some close buildings are blocked by a intermediate road, thus leading to the graph structure in Fig. 5.5(a). Detailed account is described in Section A.1.

Formally, a proximity graph is a graph – ProxG$\langle V, E\rangle$, where vertex V represents the spatial object being modeled and edge E represents the immediate proximity relationship between objects. In DT terms, any two objects that are connected by a triangle in the triangulation are linked by an edge of ProxG. Based on ProxG, the order of proximity in relation to a given enity (object or group) can be defined (see e.g. Fig. 5.5(b)).

Note that Table 5.1 only exemplifies (implicit) properties and relationships and related measurements needed by an evaluation process while implicit geometry and semantics are not included. Implicit entities (geometry and semantics) are usually added to the data via a recognition or inference process. For example, Zhang et al. [236] illustrated the use of Delaunay triangulation (identifying 'part-of' geometries) and logical reasoning to

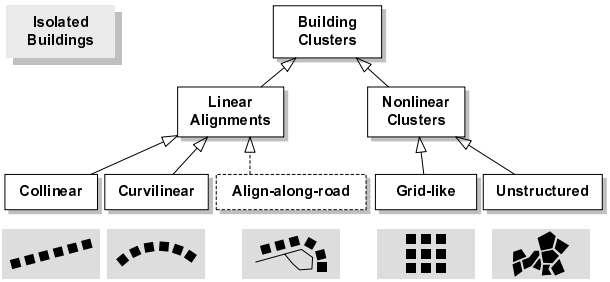


Figure 5.6 Typology and representative symbols of building patterns in topographic data

recognize peninsulas or bays in coastal areas. Lüscher et al. [114] adopted a Bayesian-based supervised classification to inference 'terraced houses' (higher-level semantics) from topographic data.

In the following sections, we first describe our attempts to recognize patterns in building clusters and to measure the properties of the structures. This is to make implicit group entities and their properties explicitly available for automated evaluation of building structures. Then, measures of spatial distribution of feature class (macro-group of objects) is developed to evaluate the global level similarities.

5.3 Patterns in building clusters

This section presents concepts and a method for detecting building patterns (clusters). Section 5.3.1 firstly defines the concept of building patterns meaningful for map generalization and quality evaluation and then proposes a typology of building patterns available on topographic maps. Based on the definition, Section 5.3.2 characterizes building patterns by their regularities. In addition, characteristics that are specific to different pattern types are also described. In Section 5.3.3, a method for building clustering is presented, and the problems of this method is shortly described in Section 5.3.4.

5.3.1 *A definition and typology of building patterns*

Building patterns are low-level and localized settlement structures which include isolated buildings with random arrangements and building patterns in clusters (Fig. 5.6). The latter is more interesting at the group level and hence the focus of this research. Semi-formally, building patterns in clusters can be defined as building groups whose elements, i.e., buildings, are homogeneous in terms of specific properties (e.g. spacing, size, orientation, shape and density), and which have typical spatial arrangements.

To distinguish between building patterns of different arrangements, we further divide building clusters into linear alignments (collinear, curvilinear and align-along-road) and nonlinear clusters (grid-like and unstructured) according to the proximity relationship between these objects. This hierarchical typology is shown in Fig. 5.6. From a spatial perspective, linear alignments appear to be more elongated and their composing buildings are arranged on a virtual or physical path (road), whereas the arrangements of nonlinear clusters appear to stretch in two dimensions but is still of high compactness (proximity). From the standpoint of geography, different types of building patterns may appear in different places depending on spatial contexts. For instance, linear alignments are common features in urban (e.g. residential areas and partly town centers), suburban and rural areas; nonlinear clusters are dominant in densely populated areas such as inner city (city centers) and commercial/industrial areas; random arrangements can be found in scattered and mountain villages [184, 186].

By studying previous literature on building structures (Section 2.1) we find that this typology is more comprehensive and generic. First, collinear alignments have been extensively studied in the literature. Second, Anders [14] presented an idea to recognize and typify grid-like building patterns, in which he proposed a typology of linear, circular, grid, star and irregular building patterns. In our approach, we refine his linear pattern into collinear and curvilinear alignments. Meanwhile, his circular pattern is a special case of curvilinear alignment in our typology (see Section 5.4.5), and the star pattern can be combined by two crossing collinear alignments. Additionally, we propose an align-along-road type which considers the relationship of building groups to surrounding roads.

The patterns in our typology can also be observed on maps (of the Netherlands, France, Spain and Britain) in the EuroSDR project [190, 192], and maps of China. Examples are shown in Fig. 5.7. It can be seen from these map examples that it is difficult to separate align-along-road alignments from collinear and curvilinear alignments, and that automated recognition of align-along-road alignments relies on the presence of road features.

Finally, we identify three abstract attributes of building patterns which put the understanding of building patterns and our recognition methods in a broader context:

Visual significance The repetition and recurrence of some selected visual elements (e.g. shape, size and color) or their combinations will lead readers to recognize significant building patterns. To emulate this

Figure 5.7 Building patterns visually identified on topographic maps: (a) curvilinear alignments, (b) A – grid-like and B – unstructured clusters, (c) collinear and (d) align-along-road alignments (Data source: Kadaster, NL)

process, our detection algorithms use Gestalt principles of perceptual organization [225]. These principles include *proximity*, *similarity*, and *good continuity* (i.e. smooth connection among compositional parts). Proximity is a fundamental principle in building clustering (Section 5.3.3). Other principles are indispensable in identifying building alignments (Section 5.4). Many geospatial processing assume implicitly a *common region* principle, as for instance buildings are usually partitioned by road networks before processing. In addition, *gradual variation* or *rhythm* is of great importance for the recognition of curvilinear alignments (see Section 5.4.3). Although this principle contradicts somewhat the concept of similarity, it remains that the pattern elements are in general similar, with gradual variations that are perceptually tolerable. It is noticeable that the repetition, good continuity and gradual variation properties are also in line with Alexander's "The Nature of Order" theory about built spaces [11].

Symbolization Symbolization is the ability to represent something (or a group) by a simple, familiar symbol through association, resemblance, or convention, and symbolization may differentiate one group from another. Topographic patterns are usually symbolized by developing templates or representative symbols that resemble their typical arrangements. Examples include radial and trellis patterns in drainage system, collinear, curvilinear, and grid-like patterns (Fig. 5.6) in settlement structures. The recognition algorithms are formalized templates for recognizing building patterns and their variations.

Ambiguity The definition and recognition of building patterns are inherently ambiguous, and the ambiguities in general take two forms. First, in the grouping process it is sometimes tricky to decide whether a building belongs to this or that group. Second, recognition methods usually make crisp distinctions between different pattern types,

though the difference may be ambiguous [53]. This means that a group of buildings could be recognized as a collinear, a curvilinear or an align-along-road alignment, depending on the person or approach involved. It may also be the case that some patterns will not be found by one algorithm and will be recognized by another (see Section 7.3). In addition, more homogenous patterns (e.g. alignments) are usually parts of general clusters. Therefore, combining different results for a more refined and complete result (see e.g. Section 5.4.6) becomes necessary. We will further discuss the involved ambiguities in Section 7.3.6.

5.3.2 *Characterization of building patterns*

According to the definition of building patterns in Section 5.3.1, building patterns in clusters can be characterized respectively by the characteristic, *homogeneity*, that is applicable to all pattern types, and the characteristics that are type-specific.

Homogeneity

Since clusters are essentially groups of objects with regular spatial distributions, we use the notion of homogeneity to characterize the degree of regularity of the composing objects in clusters. Similar ideas of homogeneity were adopted by Boffet and Rocca Serra [35], Christophe and Ruas [53], Ruas and Holzapfel [165]. Here, we calculate the homogeneity of building patterns based on the standard deviations (*STD*) of interested properties, namely spacing (nearest distance), size, orientation, and shape. The measures of these properties are described in Section 5.2. It is still possible to add extra measures in the characterization. Homogeneities of the properties (component homogeneities) are formulated as follows:

$$\mathrm{Homo}_{P_k}(\mathrm{Gr}_i) = \begin{cases} 1 - \mathit{STD}_{P_k}/\mathit{Mean}_{P_k}, & P_k \in \{\text{spacing, size, shape}\} \\ 1 - \mathit{STD}^{\star}_{P_k}/\mathit{NFactor}, & P_k \in \{\text{orientation}\} \end{cases} \tag{5.2}$$

where $P_k = \{p_1^k, \ldots, p_n^k\}$ is a vector of property values measured from a group of objects $\mathrm{Gr}_i = \{o_1, \ldots, o_n\}$. The homogeneities of spacing, size, and shape are calculated based on *coefficient of variation* (*STD/Mean*). In practice, values in P_k are assumed to follow a normal distribution and thus $\mathrm{STD}/\mathit{Mean} \in [0, 1]$ holds. In Equ. 5.2, the coefficient of variation is subtracted from 1 in order for the output value to be more easily interpreted. For example, the STD of size is 0 for a perfect regular alignment, which in turn results in $\mathrm{Homo}_{size} = 1$ using Equ. 5.2. The greater the Homo_{P_k} is less

than 1, the less regular is the alignment in terms of P_k. The denominator of the coefficient of variation (*Mean*) serves not only as a normalization factor, but also it takes the relative regularity into account. For example, if two groups have the same STD of size, the group with a larger mean of size is considered more regular, as the variation in size is relatively smaller compared with the size in the group. This can be expressed in Equ. 5.2, due to the use of coefficient of variation.

However, coefficient of variation is not suitable for calculating the homogeneity of orientation, because orientation is a cyclic variable. To normalize the homogeneity of orientation, we divide the *STD* of orientations by a factor (*NFactor*). Here we choose *NFactor* = 45°, because the max angular deviation between two regular buildings is 45° [59]. To summarize, Equ. 5.2 yields output in range [0, 1]. Note that, the *STD* of orientation in Equ. 5.2 should be interpreted and measured differently for curvilinear alignments (see Section 5.4.5 for details).

Then, the component homogeneities in Equ. 5.2 can be integrated into a single value to reflect the overall regularity of a building pattern:

$$\mathrm{Homo}(\mathrm{Gr}_i) = \sum w_k \times \mathrm{Homo}_{P_k}(\mathrm{Gr}_i) / \sum w_k \tag{5.3}$$

where w_k is the weight of homogeneities of individual properties $P_k \in \{\text{spacing, size, shape, orientation}\}$. The value of w_k can be specified by experts or obtained from machine learning. In this research, we used an equal weighting just to proof the concept. As $\mathrm{Homo}_{P_k} \in [0, 1]$, it naturally follows that $\mathrm{Homo} \in [0, 1]$.

The interpretation of output values of Equ. 5.3 is similar to that of Equ. 5.2: a higher homogeneity value indicates a higher degree of regularity, and vice versa. $\mathrm{Homo} = 1$ indicates that the group is perfectly regular, which is the case when the objects in the group are of the same size, shape, orientation, and are equally spaced (see L_3 in Table 5.2 for instance).

Type-specific characteristics

In addition to homogeneity, specific types of building patterns can be characterized in more precise ways. First of all, as meso-level group entities, building patterns should be portrayed by representative geometries (see Def. 4.3 in Section 4.1.1). Two geometry types are used to approximate the spatial extents of building patterns:

- Linear representations: skeletons of linear alignments, that is, paths connecting the centroids of the composing objects;
- Areal representations: convex hulls of nonlinear patterns; buffers of the skeletons of linear alignments.

With the above representative geometries, one can visualize and further characterize specific building patterns. For instance, collinear alignments can be characterized by *straightness* describing the sinuosity degrees of the paths and *alignment orientation* describing the orientation of the path. For curvilinear alignments, *smoothness* of the paths and *curvature description* should be emphasized. Align-along-road patterns can be separated from each other by the roads to which they align. Besides, it has an extra homogeneity, that is the homogeneity of distances to the aligned road. This property reflects the degree to which the pattern are parallel to the road. The calculation of such a distance is illustrated in Section A.1.2. On the other hand, nonlinear clusters can be characterized by their compactness (area of composing buildings divided by that of the convex hull). Grid-like patterns can be further characterized by the *orthogonality* and *parallelism* of connecting paths [14]. That is, if the buildings in a grid-like pattern are connected by rows and columns, we should get two sets of parallel lines that intersect each other approximately right-angled.

The type-specific characteristics of building patterns can be very useful in automated matching and evaluation of the patterns at scale transitions. In Section 5.4.5 we will present methods to calculate the type-specific characteristics for linear alignments.

5.3.3 *A method of building clustering*

The building clustering process is based on the idea of proximity which is modeled using Delaunay triangulation (DT) and Minimum Spanning Trees (MST) structures. In our approach, we adopt constrained Delaunay triangulation (CDT), where line segments of buildings and roads are constrained lines of CDT. To derive proximity graph ProxG (Section 5.2.4) from CDT, buildings are modeled as vertices and any two buildings (v_i and v_p) that are connected directly by at least one triangle form an $edge(v_i, v_p)$ of the graph. The implementation of CDT is refined and described in Section A.1.

Then, we derive Minimum Spanning Tree (MST) from the proximity graph. In graph theory, a spanning tree of an undirected graph G is a tree that contains all vertices of G. The weight of a tree is defined as the sum of the weights of all its constituent edges. A minimum spanning tree of G is a

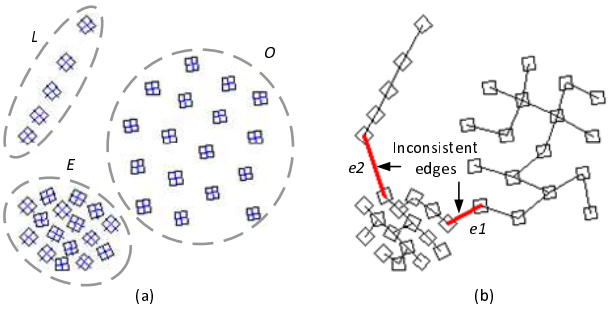


Figure 5.8 Perceptual groups *E*, *L* and *O* (a) and their MST and inconsistent edges (b)

spanning tree whose weight is the minimum among all spanning trees of G. In our approach, each MST edge is weighted by the nearest distance (Section 5.2.1) between buildings connected by this edge. To group building clusters that are perceptually significant, we adopt the notion of inconsistent edge developed by Zahn [230]. It is formally defined as follows:

$$edge_i = \begin{cases} \text{inconsistent}, & \text{if } w_i > I_l \cap w_i > I_r \\ \text{consistent}, & \text{otherwise} \end{cases} \tag{5.4}$$

where w_i is the weight of $edge_i$; I is a measure of significance that can be defined on both left (I_l) and right (I_r) side of $edge_i$:

$$I = \max\left\{f \times mean_{weight},\ mean_{weight} + n \times std_{weight}\right\} \tag{5.5}$$

In essence, we compare the weight of $edge_i$ (nearest distance between objects connected by this edge) to the mean weight of nearby edges on either side of $edge_i$ (those connected to $edge_i$ within p depth in the graph). $Edge_i$ is said to be inconsistent if its weight (w_i) is significantly larger than the mean weight of its nearby edges. Consider the criterion $I = mean_{weight} + n \times std_{weight}$, $w_i > I$ means that the weight of $edge_i$ is larger than the mean weight of its nearby edges by n units of standard deviation. The parameter n controls the significance level of the relation '*significantly larger than*'. Assume that the weights of edges in the graph follows a normal distribution, $edge_i$ with $w_i > I, n \geq 2$ is larger than at least 95% of the edges in the graph. Practically, std_{weight} may be equal to zero. In this case the criterion $I = f \times mean_{weight}$ is used to define '*significantly larger than*', where f is a magnifying factor ($f > 1$). So I is finally defined in Equ. 5.5 to integrate both criteria. The readers are referred to Zahn [230] for a detailed account of the inconsistent edges and its parametrization issues.

Building clusters are formed by pruning all identified inconsistent edges

(Fig. 5.8). The use of inconsistent edges enables reasonable inter-cluster edges to be identified, which is more adaptive than using a fixed threshold. For example, all edges in group O are identified as intra-cluster edges even if all of them are longer than the inconsistent edge $e1$ in Fig. 5.8(b). If we used a fixed threshold instead, e.g., $T = Length(e1)$, all edges in group O would have been identified as inter-cluster edges, which would lead to an unacceptable clustering for this case.

The implementation (Section 7.2) shows clustering results obtained by applications of different parameter values and to different datasets (Fig. 7.1 and 7.2).

5.3.4 *Problems of the clustering method*

First, the clustering method is based solely on distance relationship (proximity) and ignores other important factors (e.g. size, orientation, shape, etc.) that can impact the grouping of buildings. Hence the detected building groups are just general clusters, which can be further refined to obtain more homogenous groups. Second, although the clustering method seems to find clusters that also contain alignments for those well-separated cases, see e.g. Fig. 7.2(a)–7.2(c), it cannot explicitly distinguish alignments from general clusters. This means that when we see 'detected' alignments in the result, they are actually clusters in the 'eyes' of the computer. In more ambiguous situations (say patterns in dense clusters), the method gives poor clustering results, see e.g. Fig. 7.2(d)–7.2(f).

In the next section (Section 5.4) we will present our approach to explicitly recognizing and characterizing building alignments which makes use of information other than proximity and integrates aspects of computational geometry, graph-theoretic concepts and theories of visual perception.

5.4 Recognition of building alignments

Building alignments are important features and are difficult to handle in automated generalization. Commonly, nonlinear clusters are simply generalized into blocks or built-up areas due to their high densities [186]. Linear alignments, on the other hand, are usually preserved or even enhanced during the generalization, in which their forms and characteristics have to be considered [186]. In addition, alignment is one of the subject matters of the preservation constraints at the group level. Related constraints are defined in the EuroSDR project [190, 192], and in Section 6.4 we will demonstrate our approach to evaluating the preservation of building alignments in gen-

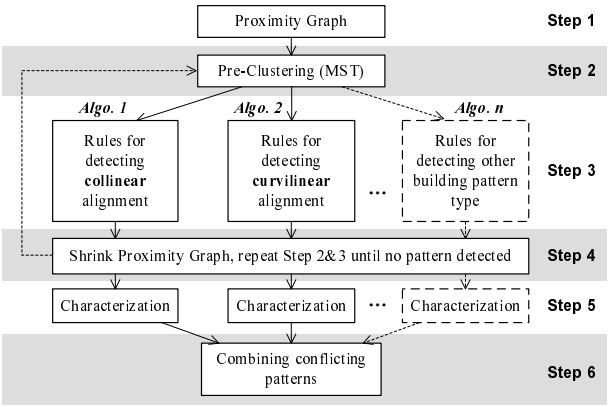


Figure 5.9
Multi-algorithm framework for building pattern recognition

eralized datasets. As a result, this section focuses specifically on automated recognition of linear alignments (especially collinear and curvilinear) to add information for the subsequent evaluation.

In the following, Sections 5.4.2–5.4.6 explain in detail some major steps of the recognition framework introduced in Section 5.4.1.

5.4.1 *Recognition framework*

The proposed framework consists of 6 steps (Fig. 5.9). It uses a multi-algorithm approach to recognize building patterns from topographic data:

1. Constructing proximity graph - ProxG$\langle V, E \rangle$ on input data: buildings are vertices and proximity relationships among objects are edges (see Section 5.2.4);

2. Performing pre-clustering based on ProxG after Step 1: MST is firstly derived from ProxG; then significant inter-cluster edges are identified and pruned from the MST (Section 5.3.3); after that, some general clusters are identified;

3. Applying different algorithms for detecting different building patterns, based on the pruned MST (see e.g. the tracing algorithm in Section 5.4.2). The tracing algorithm is adapted for each type of patterns, and employs different rules to recognize a specific pattern type (Section 5.4.3);

4. Shrinking ProxG and repeating Step 2–3 to find more patterns: after patterns have been detected, they are added to the result set and their element buildings are removed from ProxG, and then Step 2 and Step

3 are re-executed. This process is repeated until there is no pattern can be recognized by the algorithms from the remaining ProxG. For each loop, detected patterns are added to the result set (Section 5.4.4);

5. Characterizing the recognized patterns using the homogeneity measure described in Section 5.3.2. The homogeneity measure is implemented with subtle differences that are known to specific detection algorithms (Section 5.4.5);

6. Combining conflicting patterns recognized by different algorithms into a final result (Section 5.4.6).

The following sections explain the steps of the framework. These steps focus on algorithms and rules for the recognition of collinear and curvilinear alignments, each with variations of alignment-of-center and alignment-of-side, and on how the results can be combined. Steps 1 and 2 are explained in Section 5.3.3 and implemented in Section 7.2. Since the clustering process cannot ensure a persistent and meaningful grouping, we used a conservative set of parameters ($p = 2$, $n = 3$, and $f = 2$). This choice is to make sure that very significant inter-cluster edges are pruned and less significant ones are reserved for further decisions.

5.4.2 *Tracing algorithm*

This section sketches out an algorithm template (Algorithm 1), designed for tracing linear paths (i.e. a sequence of vertices that are consecutively connected by MST edges) from pruned MST. Different rules for recognizing collinear and curvilinear alignments (Section 5.4.3) are deployed as parameters in the algorithm template. Algorithm 1 starts from vertices of degree 1 or 3 and traverses the pruned MST in the following way. For each new edge encountered in the tracing iteration, the algorithm determines if the edge is consistent with the existing alignment concerning the recognition rules described in Section 5.4.3. The algorithm may find a number of edges connected with current vertex v_1, and the edge that best satisfies the rules is added to the alignment. If no edge satisfies the rules, the algorithm goes to the next edge in the searching direction (if possible) and restarts a new tracing.

After the tracing procedure, the initially detected alignments may not be perfect (e.g. Fig. 5.10). Search-for-connection and search-for-extension are required to post-process and refine the initial result. First, search-for-connection is needed in the case of a short distance between end vertices of two detected alignments (e.g. the two curvilinear alignments in Fig. 5.10).

Algorithm 1 A sketch algorithm to trace linear paths from pruned MST

Input: pruned MST; *rules* for recognizing specific alignments
Output: a collection of linear paths (LPColl)
for all vertex v_0 of degree 1 or of degree 3 in MST **do**
  $v_1 \leftarrow$ adjacent vertex of v_0
  initialize a linear path (LP): LP $\leftarrow edge(v_0, v_1)$
  repeat {add a new vertex each loop to LP if possible}
    find a vertex $v_2 \leftarrow$ adjacent vertex of v_1 and $v_2 \neq v_0$, such that $edge(v_1, v_2)$ satisfies *rules* w.r.t. LP
    if v_2 exists **then**
      add $edge(v_1, v_2)$ to LP
      $v_0 \leftarrow v_1; v_1 \leftarrow v_2; v_2 \leftarrow$ NULL
    else
      add LP to LPColl if LP contains more than 3 vertices
      reset LP: LP $\leftarrow edge(v_1, v_2)$
      $v_0 \leftarrow v_1; v_1 \leftarrow v_2; v_2 \leftarrow$ NULL
    end if
  until v_1 is of degree 1
  add LP to LPColl if LP contains more than 3 vertices
end for
Post-processing: search-for-connection and search-for-extension

This search can be done by connecting the two end vertices of both alignments, if the connecting edge does not violate any of the recognition rules used. Second, search-for-extension is required because MST only connects locally shortest edges rather than visually desired edges. In Fig. 5.10, the two white buildings on the bottom right should have been grouped into the alignment of the four dark blue buildings. However, MST does not connect the two white buildings to the four because of the intermediate non-aligned building. Consequently, the tracing procedure stops at the sharp turn. To solve this problem, we then extend each traced alignment at its end vertices and search for adjacent edges in proximity graph instead of MST. If an edge is found that best satisfies the required rules, the edge (and its connecting

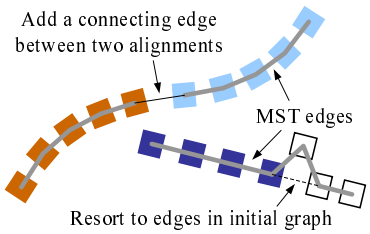


Figure 5.10 Post-processing of initially detected alignments

buildings) is added to the alignment.

5.4.3 *Recognition rules*

This section addresses the recognition rules for collinear and curvilinear alignments. For both alignments, alignment-of-center and alignment-of-side techniques are presented. The former technique uses default rules, while the latter technique is described in the section on 'ONE-SIDE ALIGNMENTS'

Collinear alignments

A salient feature of collinear patterns is that buildings are aligned as if they form a straight line. The straightness is controlled by the path angle of alignments. Path angle at vertex v_i is the angle formed by $vector(v_{i-1}, v_i)$ and $vector(v_i, v_{i+1})$ in the path (Fig. 5.11(b)).

The detection of the collinear pattern requires proximity, path angle, orientation and size similarity rules. Shape similarity is ignored because after studying topographic maps, we found that shape is not a dominant feature in determining a collinear alignment. First, most buildings have similar man-made shapes. Second, buildings with complex shapes are usually much bigger than the ordinary ones, so complex buildings can always be identified by size similarity rule. Nevertheless, shape property is still used for characterizing the detected patterns (see Section 5.3.2).

Proximity rule is again based on the notion of inconsistent edge introduced in Section 5.3.3. That is, if the weight of a new edge encountered in Algorithm 1 is inconsistent with respect to the existing alignment, the edge is regarded to be not part of this alignment. **Path angle rule** requires that angular deviations between the new edge and all edges in the existing alignment should be within certain threshold (*PathAngleDev*). This rule suppresses accumulative angular errors and ensures a straight-line-like alignment. **Orientation similarity rule** (*OrientationDev*) ensures that the angular deviation between any two adjacent buildings does not exceed 45°, as according to Duchêne et al. [59]. **Size similarity rule**, then, prevents the size contrast (*SizeContrast*) between any two adjacent buildings in the alignment from being too large (i.e. bigger building/smaller building $<$ *SizeContrast*).

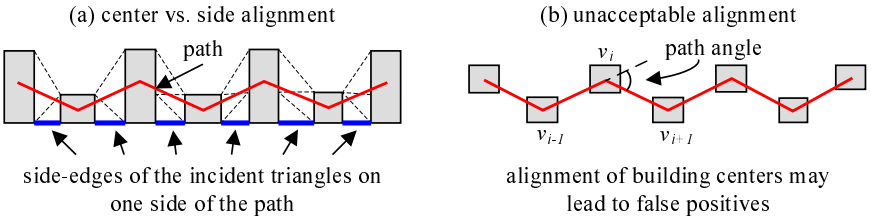


Figure 5.11 Alignment-of-center vs. alignment-of-side technique

One-side alignments

Some collinear alignments (see Fig. 5.11(a)), which can be widely found in real topographic data and are also perceptually significant, cannot be detected by the above model without an adaptation of parameters. This is mainly due to the varying size of adjacent buildings (Fig. 5.11(a)) that increases the path angles, which leads to a violation of the path angle rule.

To detect such patterns, we use alignment-of-side instead of alignment-of-center technique to measure the alignment of buildings. The alignment-of-side technique is facilitated by the incident triangles stored after the construction of CDT (e.g. the dashed triangles in Fig 5.11(a)). To implement this technique, new path angle is calculated as the vector angle between adjacent side-edges of the incident triangles (e.g. the bold blue lines in Fig. 5.11(a)). Note that the new path angle can be measured on two sides of an alignment. If this new path angle rule, proximity and orientation rules are satisfied on at least one side, the group is recognized as a one-side alignment.

This alternative better resembles the way how human eyes perceive one-side alignments. In particular, it is more robust in detecting one-side alignments than arbitrarily adjusting *PathAngleDev*, as the latter will increase the chance of false positives. For instance, using alignment-of-center technique, the one-side alignment in Fig. 5.11(a) can be detected by increasing *PathAngleDev*. However, this will identify the group in Fig. 5.11(b) as a collinear alignment, which is unacceptable. Note that the alignment-of-side technique can also be applied for curvilinear alignments. This technique does not address size similarity rule as it is not required for one-side alignments (e.g. Fig. 5.11(a)). A mechanism for combining alignment-of-center and alignment-of-side techniques is presented in Section 5.4.6.

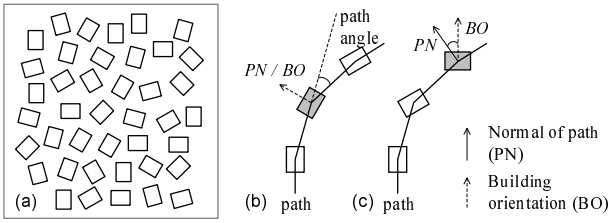


Figure 5.12 Rules for human eyes to recognize curvilinear patterns (after [40])

Curvilinear alignments

Curvilinear building alignments are patterns with gradual variations. Empirically, paths formed by such alignments gradually alter their heading angles and orientations of composing buildings, while keeping other properties such as spacing and size as similar as possible. Because most of the meandering features represented in spatial databases are too complex to be described by mathematical functions, we use the theory of 'association field' [67] to formulate rules for recognizing curvilinear alignments. This theory of visual perception suggests a set of localized linking rules that explains why and how curved paths can be detected by human observers (Fig. 5.12). Contemporary researchers in visual perception also acknowledged that these rules are more concrete and realizable than Gestalt principles [40, p. 159].

In general, the psychological experiment by Field et al. [67] suggests that the human perception of curved or undulated paths formed by adjacent visual stimulus (e.g. Fig. 5.12(a)) is reflected by good continuity (smooth connection). More specifically, it depends on two rules. These are the smooth path and the degree to which the path elements (e.g. composing buildings in our case) are aligned or misaligned along the path. Take Fig. 5.12 for example, if the normal vector of the path (PN) (Section 5.2.2) coincides with the building orientation (BO), the building is perfectly aligned (the shaded building in Fig. 5.12(b)). If PN and BO are far apart, the building is misaligned (the shaded building in Fig. 5.12(c)). Field et al. [67] show further that the allowed maximum path angle (*MaxPathAngle*) is 40°–60°, provided that the elements are aligned along the path; misalignment by ±15° reduces the performance of detecting such paths (see also [40]). These rules explain why human visual system can see a 'snake' in Fig. 5.12(a).

We formalize the above rules as follows. First, $MaxPathAngle \in [40°, 60°]$ was selected for **smooth path rule**. It means that the path can be regarded as smooth if its path angle is smaller than *MaxPathAngle*. Second, **misalignment rule** specifies an allowed deviation between the BO of a building

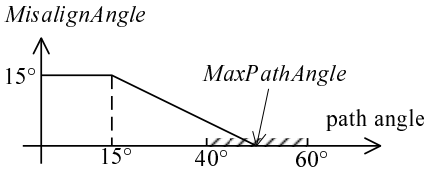


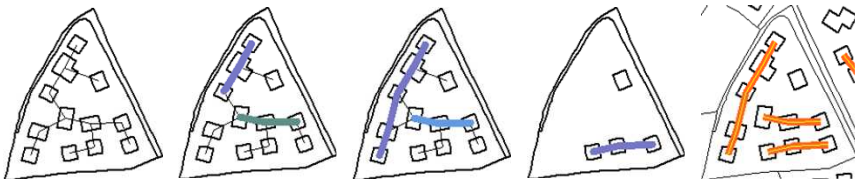
Figure 5.13 The function describing the relationship between *MisalignAngle* and calculated path angle

and the PN at this building (*MisalignAngle*). *MisalignAngle* is a dependent variable of the path angle at the building. The relationship between *MisalignAngle* and the path angle is that *MisalignAngle* is relatively relaxed ($= 15°$) when the path angle is relatively small ($\leq 15°$), and *MisalignAngle* then decreases with the increase of the path angle, and it ends with zero when the path angle reaches *MaxPathAngle* (Fig. 5.13). If the calculated misalignment angle of a building exceeds its *MisalignAngle*, the building is regarded as a misaligned building.

The detection of curvilinear building patterns is based on Algorithm 1 specialized by the above formalized rules, along with proximity and size similarity rules. The alignment-of-side technique as presented in the last section is also applicable for curvilinear patterns.

5.4.4 *Improvement by repetition*

This section explains how the post-processing in Algorithm 1 works with the repetition step (Step 4) in the recognition framework, in order to improve the recognition quality. Take the algorithm adapted for collinear alignments with alignment-of-center for example, Fig. 5.14(b) shows the detected collinear alignments without post-processing (Section 5.4.2) after the algorithm has been executed once. By applying the post-processing (e.g. search-for-extension) the result was improved (a longer alignment was detected in Fig. 5.14(c)). Visually, the alignment on the bottom of Fig. 5.14(c)



(a) Pruned MST (b) Initial result (c) Extension (d) 2^{nd} round (e) Final result

Figure 5.14 Repetition of intermediate steps to improve the recognition quality (Data source: Kadaster, NL)

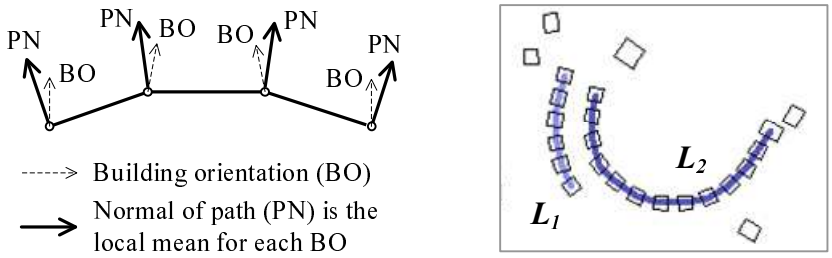


Figure 5.15 Using local PN instead of the mean of BO to compute the homogeneity of orientation for curvilinear alignments

should have been recognized, but it was not detected due to the structure of MST (Fig. 5.14(a)). As described in Section 5.4.1, the framework automatically records the detected alignments and then removes them from the proximity graph, after which MST is recalculated and the above-mentioned steps are re-executed. This iteration proceeds until all possible alignments are successfully recognized (e.g. Fig. 5.14(d) and 5.14(e)).

5.4.5 *Characterization of building alignments*

The building alignments detected by specific algorithms can be characterized using the homogeneity measure introduced in Section 5.3.2. As also having been mentioned in Section 5.3.2, the *STD* of orientation should be interpreted and measured differently for different pattern types.

For collinear alignments, it has a standard calculation: $STD = ((BO_i - Mean(BO))^2/N)^{1/2}$, where BO represent building orientation (see Section 5.2.2 for its calculation). However, *STD* requires a different interpretation for curvilinear alignments. In essence, *STD* describes the degree of deviation from mean. To better describe the homogeneity of orientation for a curvilinear alignment, normal of path (PN) at each vertex (see Section 5.2.2 for its calculation) is more appropriate as the local mean for each BO, instead of the real mean of BO (Fig. 5.15). In particular, when all BOs align themselves fully with their local PNs, the buildings are regarded to be perfectly aligned in terms of orientation. As a result, *STD* of orientation (starred in Equ. 5.2) for curvilinear alignments is better calculated by the formula: $STD^\star = ((BO_i - PN_i)^2/N)^{1/2}$. Note that the wall statistical weighting approach (Section 5.2.2) yields two perpendicular directions for each building, of which the one that is cross the path is used here (Fig. 5.15).

In addition to the measure of homogeneity, sinuosity and curvature can be

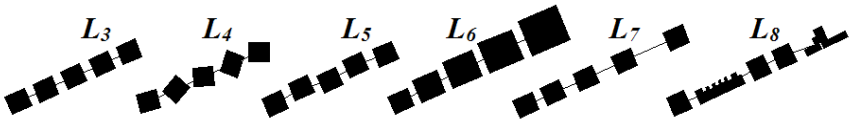


Figure 5.16 Artificial dataset: a perfectly aligned collinear pattern L_3 and its variations

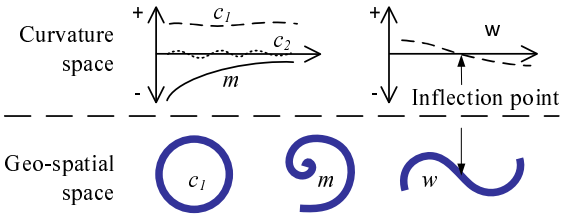


Figure 5.17 Hypothetical distributions of different forms of alignments

used to further characterized linear alignments. For any alignment A_i with representative geometry $\{v_1, \ldots, v_n\}$, sinuosity (inverse of straightness) can be estimated as: $Sinuosity(A_i) = 1 - Dist(v_1, v_n)/\sum_1^{n-1} Dist(v_i, v_{i+1})$. The index $Sinuosity(A_i)$ follows within [0,1]. $Sinuosity(A_i) = 0$ indicates A_i is perfectly straight; with increase of this index, the alignment A_i becomes more sinuous (see e.g. Table 5.2).

Mathematically, the curvature (κ) of a plane curve at any point (p) is defined as a local change (derivative) of the unit tangent vector (T) with respect to arc length (s): $\kappa = \lim \frac{\delta T}{\delta s}$, where s is the distance from the point after the local change (p′) to p. For curves stored in spatial databases (i.e. polylines), local curvatures for every vertex (v_i) can be estimated as: $LocalCurvature(v_i) = PathAngleDev(v_i)/Dist(v_{i-1}, v_{i+1})$. The denominator is roughly constant for a given alignment so we use changes of path angle to analyze local curvatures: $LocalCurvature(v_i) \propto PathAngleDev(v_i)$. Also in this way, the estimation becomes easier to interpret. Finally, it is possible to analyze forms of alignments, e.g., arc-like, circle-like, based on the distribution of estimated local curvatures (Fig. 5.17).

Fig. 5.17 illustrates three typical forms. They are alignments with constant curvature (c_1, c_2), with monotone curvature (m) and with wavy curvature (w). Specifically, if the curvatures of an alignment almost constantly distribute around the zero (c_2 in Fig. 5.17), it has a straight-line-like shape. If the curvatures constantly distribute with a considerable distance to the horizontal axis (i.e. zero), the alignment has an arc-like shape. Further, if the head and tail of an alignment with constant local curvatures are close to

ID	STD/Mean			STD*	Sinuosity	Homo
	Spacing	Size	Shape	Orientation		
L_1	0.07	0.00	0.000	0.06°(STD*)	0.045020	0.97
				17.21°(STD)		0.89
L_2	0.08	0.10	0.000	0.08°(STD*)	0.362876	0.93
				25.94°(STD)		0.81
L_3	0.06	0.00	0.000	0.00°	0.000048	0.99
L_4	0.14	0.01	0.002	22.63°	0.000056	0.84
L_5	0.12	0.00	0.000	0.00°	0.006516	0.98
L_6	0.05	0.43	0.001	0.00°	0.000213	0.88
L_7	0.50	0.00	0.000	0.00°	0.000014	0.88
L_8	0.13	0.18	0.283	0.40°	0.001837	0.85

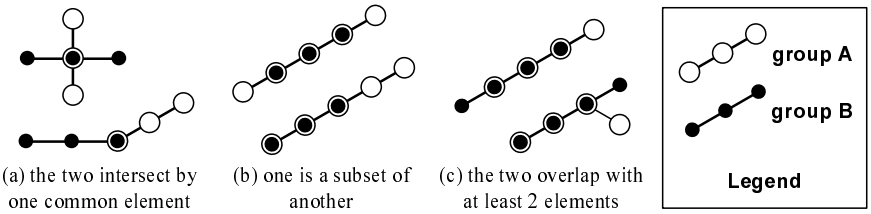
Table 5.2 Characterization of sample alignments $L_1 - L_8$

each other, the alignment forms a circle (c_1 in Fig. 5.17). Wavy alignments have the property that its curvatures go from positive to negative part of the curvature space or in an opposite direction; crossing points at horizontal axis identify inflection points in the alignments (e.g. w in Fig. 5.17). Alignments with Monotone increase or decrease in curvatures without changing signs are only theoretical (e.g. the spiral m in Fig. 5.17).

Table 5.2 demonstrates component homogeneities (STD/Mean) and integrated homogeneity (Homo) for sample alignments ($L_1 - L_8$ in Fig. 5.15 and 5.16). The regularity of orientation is represented using STD* which is more intuitive. When aggregating the component homogeneities into a single value (Homo), we divide STD of orientation by 45° to normalize it. Table 5.2 shows that the adapted STD* better captures our perception on the regularity of orientation for the curvilinear alignments (L_1 and L_2) than does the standard STD (values indicated in gray). Actually, L_1 and L_2 are curvilinear alignments of (nearly) perfect regularity. The characterization of the alignments ($L_3 - L_8$) gives an intuition how the change of spatial arrangement and geometric properties alters the values of component homogeneities (changed properties are indicated in gray). Examples of curvature descriptor are shown in Section 7.3.4 for linear alignments.

5.4.6 *Combining conflicting patterns*

As shown in Section 7.3, the algorithms described in previous sections yielded reasonable results, but the results from different algorithms were not in itself complete. For instance, although the curvilinear algorithm is



(a) the two intersect by one common element (b) one is a subset of another (c) the two overlap with at least 2 elements

Figure 5.18 Relationships between two potentially conflicting building patterns

able to recognize some collinear alignments, it does not recognize all of them successfully. In other cases, a group of buildings may at the same time be recognized as e.g. a collinear and a curvilinear alignment. As a result, those potentially conflicting patterns have to be combined and the most probable one should be selected for a better and more complete result.

The basic criteria for combining conflicting alignments are the number of elements (*NumOfElem*) contained in building patterns and their homogeneity values (i.e. *Homo* in Equ. 5.3). The more elements a pattern contains the better it is; the higher the homogeneity value the stronger the pattern. According to these two principles, we formulate a more detailed scheme to combine two potentially conflicting patterns (suppose groups A and B in Fig. 5.18 are patterns recognized by different algorithms, e.g., collinear and curvilinear ones):

1. If groups A and B share one common element (e.g. Fig. 5.18(a)), rather than treating them as combined patterns, we keep both patterns as individual ones; otherwise the combination can yield rather complex and irregular groups;

2. If one group B is the subset of another group A (e.g. Fig. 5.18(b)), group A is selected as the final result. In a special case where A and B are exactly the same but they are assigned to different pattern types (by different algorithms), the one with a higher *Homo* value is selected;

3. If two groups A and B have at least two elements in common (e.g. Fig. 5.18(c)), we consider the significance creterion in Equ. 5.6. The group with a higher value of the criterion is selected;

$$Significance(Gr_i) = Homo(Gr_i) \times NumOfElem(Gr_i) \tag{5.6}$$

4. For cases where the patterns are detected by only one of the algorithms, they are added to the final result set.

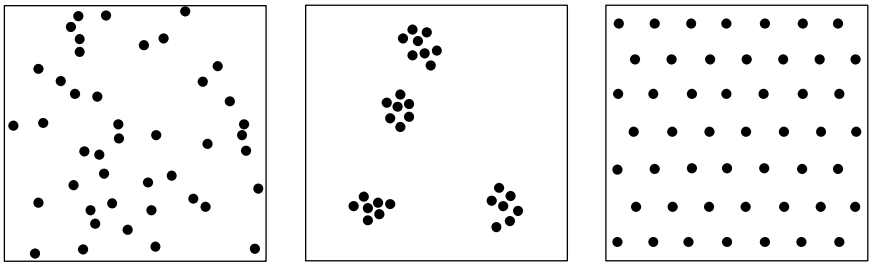


Figure 5.19 Fundamental types of point distributions: (left) random, (middle) clustered, and (right) regular pattern (adapted from [36])

The combining process should be applied in two stages in the framework. The first stage combines the alignments of the same type of pattern recognized by alignment-of-center and alignment-of-side algorithms. This stage produces a harmonized result for each pattern type. The second stage is to evaluate and combine the results of different pattern types (e.g. collinear and curvilinear) that are already harmonized, which leads to a final result.

The implementation and experimental results of the presented recognition framework are reported in Section 7.3. The recognition results are validated by cartographic experts in Section 7.3.3 using the expert survey in Appendix B and discussed in Sections 7.3.5–7.3.7.

5.5 Measures of Spatial Distribution

Spatial distribution is a structural characteristic that describes the distribution of spatial objects across a two dimensional space (e.g. a meso-/macro-level region on a map). It is essentially about the distribution of locations and differs largely from the distribution of (geometric or semantic) properties over a designated map extent. The latter can be analyzed using standard statistical tools such as distribution analysis (see e.g. [163, 164]).

The spatial distribution (pattern) can be characterized into three fundamental types, namely random, regular (uniform), and clustered pattern [36, 95]. This has been investigated extensively and systematically in the field of spatial/geo-statistics and spatial point process/patterns (see Fig. 5.19). Note that spatial distribution is a kind of global pattern; whereas the building patterns and alignments (Sections 5.3 and 5.4) are a kind of local pattern, where the spatial distribution of the composing buildings is characterized to be regular (or homogeneous).

Spatial autocorrelation

Spatial autocorrelation is commonly used to measure spatial distributions [74, 147, 226]. It usually reflect to what extent is the distribution random, regular, or clustered. However, indicators of spatial autocorrelation are mostly designed for raster data, point patterns, or the patterns of variables associated with adjacent geographic units (e.g. administrative boundaries), and the indicators have various limitations [51, 52, 74, 177] (see Section 5.5.2).

Entropy-based techniques

In the geospatial domain, several measures have also been proposed and tested to quantify the spatial distribution of maps. For example, many researchers [31, 87, 110] measure spatial distribution using entropy-based techniques adapted from information theory [174]. However, these measures only reflect the complexity of maps in terms of information content, and their values cannot be used to indicate whether the distribution is random, regular, or clustered.

Density and spatial distribution

Spatial distribution can be indicated by the distribution of densities perceived over a two dimensional map space. For example, if a group of objects is of clustered pattern, its density distribution should also be clustered (nearby objects have similar density values), and vice versa. This has the advantage that, besides the qualification of spatial distribution into random, regular, or clustered patterns, spatial distribution is transformed into a two dimension space populated with density values (density map). Density maps can thereby be measured more quantitatively. As a consequence, this requires that the distribution of densities has to be measured appropriately (Section 5.5.1).

Cartographic constraints related to the measure of density are mentioned in the AGENT project [1] for instance to keep the black-and-white (ink-to-paper area) ratio and density balance over the map space. In spatial point process/pattern terms, map density is also referred to as *intensity* of the point process.

This section briefly describes how to estimate densities of individual objects distributed over a map space (Section 5.5.1) and how to qualify spatial distribution using measured densities combined with a spatial autocorrelation technique (Section 5.5.2). This section ends with a discussion of the

applicability of density-based spatial autocorrelation to quality evaluation (Section 5.5.3).

5.5.1 *Measure of density distribution*

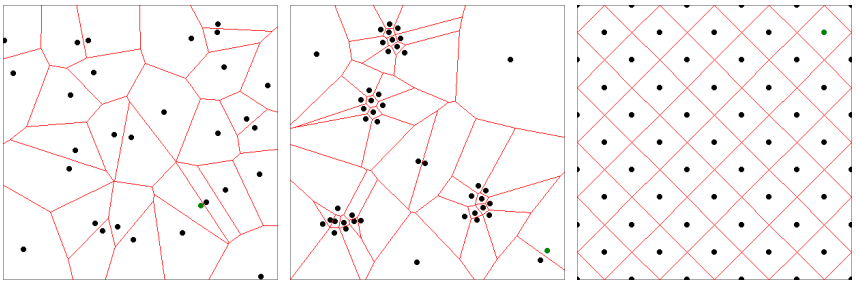
Constraints related to density can be categorized into two types. The first type of density is defined in common sense by ink-to-paper area ratio (area of symbols per unit area). The second type of density is the balance/difference or distribution of feature densities across a map which should be retained during generalization. According to [1], the first type is a structural constraint (e.g. 'buildings with densities higher than 0.1 should be changed into built-up areas') while the second is a *Gestalt* one. The *Gestalt* density constraint addresses for instance that the differences in densities of scattered objects should be retained or stressed according to different thematic classes, in order to reflect the characteristics of the reality [184]. This section will describe an approach to measuring the distribution of densities.

Partitioning of map space

Four types of spatial partitions are identified to divide up the map space for density calculation (some of which are also discussed in Section 4.1.2):

- Map extent or convex hull of all the objects
- Feature-based partitioning
- Spatial tessellations
- Individual object-based

These four partitions analyze spatial densities at different levels of detail. First, density calculated using map extent or convex hull is the roughest one and only makes sense at global level, which is not of much help to describe the spatial distribution across the map. Second, feature-based partitioning can be used to compute differences in density between different partitions, and it gains deeper insights into spatial distribution density. For example, Ruas [162, 164] discussed the evaluation of block density change before and after generalization. Third, Brazile [38] also proposed an evaluation model based on the partition formed by road and street to reduce the computation effort. However, this approach has two major problems. On the one hand, partitioning features are not always available in every situation (e.g. roads in rural areas are not always enclosed). On the other hand, the measured density of a partition is still not fine-grained. Grid-based partitioning for



(a) Random pattern (b) Clustered pattern (c) Regular pattern

Figure 5.20 Impact areas of points based on Voronoi diagram (red)

density computing was studied by several authors. Jansen and van Kreveld [96] for instance used arbitrary grid to evaluate the consistency of map generalization. However, one has to choose the size of grid upon which the density estimation can be influenced greatly [37].

To overcome the above-mentioned problems and to provide fine-grained density information, we propose to calculate densities based on individual objects, which is essentially a contextual operation.

A definition of local density

To define local densities of spatial objects, we have the following observation:

Observation 5.1. *When looking at a map of a settlement with a random or clustered distribution of buildings, we find that the distribution of densities across the map space is heterogeneous, i.e., differences in densities are obvious; the closer the buildings are clustered, the smaller their impact areas (space surrounding each building) and thus the higher their local densities are, and vice versa. A settlement of buildings with a regular (uniformed) distribution induces a map of homogeneous distribution of perceived densities, i.e., densities are more or less the same across the map space.*

The *impact area* of a spatial object is formally defined as the area that any point in the area are closer to this object than to other objects. It is a concept similar to the Voronoi diagram for point features (see e.g. Fig. 5.20). However, the concept is harder to implemented for linear and areal geometry types. We therefore propose to use a Voronoi-like diagram (Section A.2) to approximate the impact areas of linear and areal features.

According to the above observation, we now know that local density is in inverse proportion to the size of the impact area. That is, the density of an object increases as its impact area becomes smaller. This reflects that local density is a contextual measure as the value is influenced by an object's neighborhood. Finally, by incorporating the metric information (e.g. length and size), we define local density for point, linear, and areal features in the following way:

$$Density(\mathrm{P}) = 1/Area(ImpactA(\mathrm{P})) \tag{5.7}$$

$$Density(\mathrm{L}) = Length(\mathrm{L})/Area(ImpactA(\mathrm{L})) \tag{5.8}$$

$$Density(\mathrm{A}) = Area(\mathrm{A})/Area(ImpactA(\mathrm{A})) \tag{5.9}$$

where P, L, and A represent point, linear, and areal geometry types respectively; $ImpactA(\cdot)$ represent the impact area of some object.

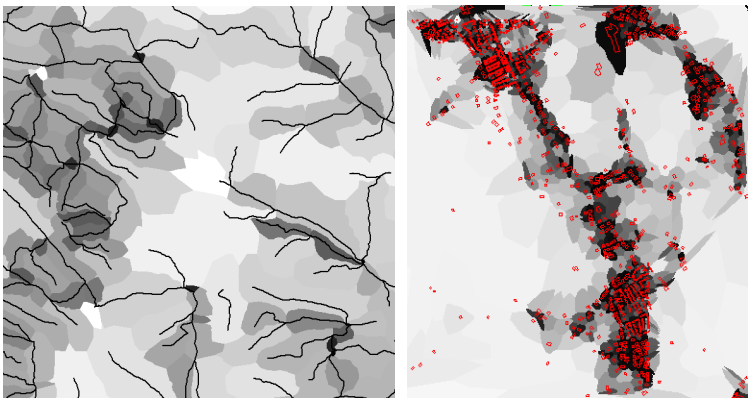
Measure of local densities and density distribution

The core of the measure of local density lies in the notion of *impact area*. In this research, we proposed a Voronoi-like diagram to mimic the function of the Voronoi diagram of points to approximate the impact area for linear and areal features. The Voronoi-like diagram is generated by skeletonizing the gap space among map objects, which is again based on Delaunay triangulation (see Section A.2 for details). Based on the so-called Voronoi-like diagram, We are able to realize the impact areas of linear and areal features. Fig. 5.21 illustrates calculated densities for a network and a set of buildings based on the Voronoi-like diagram using Equ. 5.8 and 5.9. The density distribution is portrayed on a two-dimensional space and can be used to further study the spatial distribution of the original data.

The local density has the advantage that it is object-oriented and fine-grained, meaning that local densities can be stored as an attribute of individual objects. Besides, the partition of map space naturally captures the way how proximate objects compete for space and reflects better how densities are distributed. In the next section, we adapt a spatial autocorrelation index (Moran's I) to measure the global spatial distribution of point, linear, and areal features based on the local density measure.

5.5.2 *Density-based measure of spatial autocorrelation*

Spatial autocorrelation can be used to characterize global pattern (random, clustered, regular) of a spatial distribution. However, current statistical



(a) Road density (b) Building density

Figure 5.21 Distribution of local densities measured for network and areal features

techniques were developed for special purposes and hence require *ad-hoc* forms of input data. For example, the Nearest Neighbor Index (NNI) and Ripley's K can be used to characterize the point patterns [36, 183], but the value of NNI depends on the size of extent defined for the points [161]. Another statistic is the quadrat analysis of points, which divides the study area into a regular grid. Nevertheless, Chou [52] shows that this method is not well suited for analyzing point patterns, as the method is unpredictable when the size of the grid is changing. This is also noticed by Rogerson [161] who found that the quadrat method only yields promising results when an optimal grid is determined.

For the spatial patterns of variables associated with adjacent polygonal units, commonly used statistics include Moran's I, Geary's C, joint count analysis, etc. Moran's I is, among these others, the most widely used index of spatial autocorrelation [52]. The limit of original Moran's I is that it can only be applied to values assigned to spatial partitions (e.g. categorical/polygonal, land-use maps, and values summarized by administrative units). The index can hardly be applied to discrete data as is commonly the case in spatial databases.

To summarize, the above-mentioned indicators are not well adapted for analyzing the spatial distribution of maps consisting of discrete objects (e.g. points, curves, polygons, and networks). In the following, we adapt Moran's I to extend the index to better suit for common data types in spatial databases.

Moran's I coefficient

As mentioned above, Moran's I coefficient is widely used in quantifying spatial autocorrelation in the literature. We will now introduce how to calculate the I coefficient based on the above-mentioned materials [51, 52, 74, 183]. The Moran's I is defined as follows:

$$I = \frac{N \sum_i \sum_{j,\, i \neq j} W_{ij}(Y_i - \overline{Y})(Y_j - \overline{Y})}{(\sum_i \sum_{j,\, i \neq j} W_{ij}) \sum_i (Y_i - \overline{Y})^2} \tag{5.10}$$

where N is the number of the variables Y_i associated with spatial units indexed by i in the study area; Y_i and Y_j are the values of Y associated with unit i and j respectively; $\overline{Y}$ is the average of Y; W_{ij} is a weighting function measuring the spatial contiguity of locations i and j. Spatial units in Moran's I are usually represented by polygons that are topologically adjacent to each other, e.g., raster cells and administrative boundaries.

The above I coefficient is a global index. That is, the index evaluates whether the global spatial distribution is random, clustered, or regular. Moran's I ranges from -1 to 1, where $I = 1$ indicates a positive autocorrelation and implies a complete clustered pattern, while $I = -1$ indicates a negative autocorrelation and implies a regular (scattered) pattern. When the value is not significantly different from zero, there is no autocorrelation: the variables associated with spatial units is completely random [51, 52]. Note that spatial units must be partitions of the study area [52].

Specifying the variable and weighting function

First, we use voronoi-like diagram (Section A.2) as the spatial partition in calculating Moran's I. More specifically, either areas of partition units (impact areas) or local densities are specified as variable Y in Equ. 5.10.

Another important issue is the determination of the weighting function (W_{ij}) in Equ. 5.10. First-order topological contiguity is widely used to specify W_{ij} for calculating spatial autocorrelation [see e.g. 74]:

$$W_{ij} = \begin{cases} 1 & \text{if i, j are first-order neighbors} \\ 0 & \text{otherwise} \end{cases} \tag{5.11}$$

The first-order neighbor relationship is defined based on the proposed Voronoi-like diagram. That is, the first-order neighbor for points are represented by adjacent Voronoi regions, and linear and areal features by adjacent Voronoi-like regions (see e.g. Fig. 5.22). Other options to define W_{ij} include

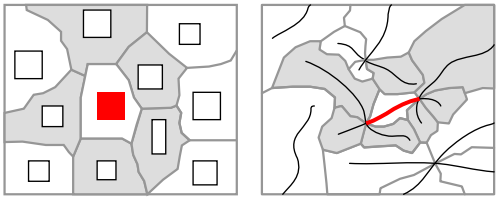


Figure 5.22 1st-order neighbor for areal (left) and linear network (right) features: the units in gray are the 1st-order neighbors of the target objects (red)

area, common boundary length, distance (e.g. $\frac{1}{d^2}$), higher-order neighbor (see e.g. Fig. 5.5(b)), and their combinations [52, 74].

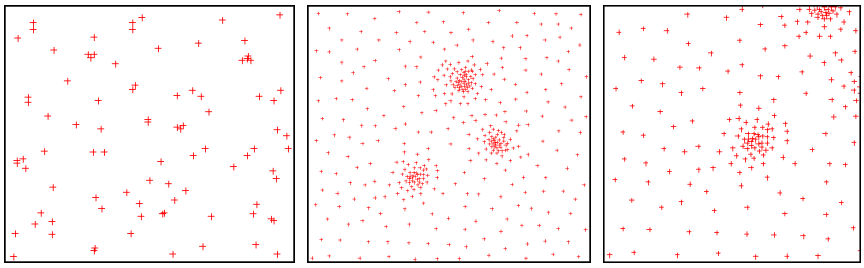
5.5.3 *Applicability to quality evaluation*

Applying the adapted Moran's I to different datasets shows that it is able to characterize the spatial distributions of points, curves (e.g. transportation networks) and disjoint polygons (e.g. buildings) in a uniform way. The experimental results can be found in [237]. The proposed method is able to distinguish precisely and objectively between random and clustered patterns for all the three types of geometries.

However, this adapted Moran's I is not adequate for comparing the similarity between different distributions for the evaluating whether spatial distribution is kept or not. This is illustrated by applying the index to the point patterns in Fig. 5.23. Although the calculated I values confirm their hypothetical patterns, the index cannot distinguish qualitatively similar but quantitatively different distributions. For example, distributions (b) and (c) are rather different, but the measured values indicate that they are similar. This is reasonable since spatial autocorrelation is a global approach which only distinguishes between degrees of randomness or clustering. In this respect, the calculated values for (b) and (c) in Fig. 5.23 indicate only that they have the same degree of clustering. For the purpose of evaluating the similarity of spatial distributions, more expressive measures have to be sought.

5.6 Similarity between spatial distributions

As discussed in Section 1.2.2, Chapter 2 and Section 3.4, preserving the spatial distribution of groups of objects is a needed feature class level constraint for a meaningful evaluation. Previously this constraint was difficult to formalize mainly due to a lack of measure that compares the similarity between two spatial distributions. Like the concept of similarity in computer vision and image processing, similarity measures should give similar



(a) random (b) clustered (c) clustered

Figure 5.23 Point patterns with hypothetical distributions; calculated I values: 0.01 (a), 0.74 (b) and (c)

descriptions to similar spatial distributions, and different descriptions to distributions that are dissimilar. In this respect, measures of spatial distribution described in Section 5.5 are inadequate, because the measure fails to distinguish between two different spatial distributions.

However, similarity between spatial distributions has requirements different from domains of computer vision and image processing. In essence, similarity descriptors in computer vision are about the similarity of shapes of individual objects, and usually they have a set of properties that are invariant under transformations such as translation, scaling, and rotation [146]. In map generalization, on the other hand, similar spatial distributions become dissimilar if one of the distribution is displaced, rotated, or scaled. It is essentially about the similarity of locations of groups of objects where metric differences are crucial in defining the similarity. As a result, instead of developing a measure that describes the unique characteristics of the spatial distribution of locations, this section proposes a relative measure that calculates the similarity between two distributions of locations. Section 5.6.1 presents the measures and Section 5.6.2 demonstrates the expressive power of the measures. The applicability of the proposed measures to quality evaluation is discussed in Section 5.6.3. In Section 7.6.2 the proposed measures are applied to evaluate a constraint on keeping spatial distribution of buildings.

5.6.1 *Distance-based similarity measure*

Consider two non-empty set $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_m\}$, where $a_i, b_i \in \mathbb{R} \times \mathbb{R}$ are locations of two dimensional space, the similarity between these two spatial distributions is described in the following.

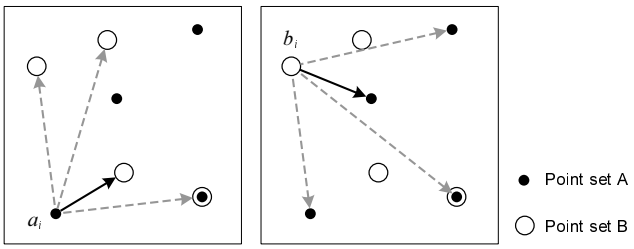


Figure 5.24
Nearest distance between a_i and set B (left) and between b_i and set A (right); solid arrows indicate nearest distances

Nearest inter-set distances

To begin with, we define the nearest distance of a location $a_i \in A$ to set B as:

$$ND(a_i, B) = \inf_{b_j \in B} d(a_i, b_j) \tag{5.12}$$

$ND(b_i, A)$ can be given in a similar way. The idea is demonstrated in Fig. 5.24. For the set of points A, we can derive a set of nearest distances between every points in A and set B: $NDS_{AB} = \{ND(a_i, B) | a_i \in A\}$. Likewise, $NDS_{BA} = \{ND(b_i, A) | b_i \in B\}$ can be derived for set B. The nearest inter-set distance (*NISD*) between A and B can be defined based on the above sets of nearest distances:

$$\begin{aligned} NISD(A, B) &= F(NDS_{AB}) \\ NISD(B, A) &= F(NDS_{BA}) \end{aligned} \tag{5.13}$$

where $F(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}$ represents a family of summarizing functions (e.g. mean, square mean, standard deviation) that maps a set of real values into a single value. To be more generic, we use weighted form of summarizing functions such as weighted mean (F_1) and weighted square mean (F_2), where weights are assigned to points in set A and B to model their relative importance in relation to other points in the same sets.

Given two non-empty sets of objects A and B, the maximum *ND* should not exceed the diagonal length of the union of the bounding boxes of A and B (*MaxND*). Based on this *MaxND*, we can conclude that *NISD* is also bounded, with $NISD \in [0, MaxND]$ if weighted mean is used as the summarizing function and $NISD \in [0, MaxND^2]$ if weighted square mean is used.

Several properties hold for NISD:

1. $NISD(A, B) \geq 0$ (non-negativity)
2. $NISD(A, B) \neq NISD(B, A)$ (non-symmetry)

3. $NISD(A, B) = 0$ if and only if $A \subseteq B$
4. $NISD(A, B) = NISD(B, A) = 0$ if and only if $A = B$

First, non-negativity is obvious. *NDS* is a set of positive real numbers, and thus the summation of positive values should not yield a negative value. The second property is also evident since set A and B are in general different and therefore NDS_{AB} is different from NDS_{BA}. Third, if $A \subseteq B$ then $ND(a_i, B) = 0$ and NDS_{AB} is a set of zeros. The summation of a set of zeros is still zero. The fourth property can be inferred from the third property.

Similarity measures

To fully describe the similarity between two spatial distributions, both $NISD(A, B)$ and $NISD(B, A)$ should be considered. This is because the two nearest inter-set distance measures are general different (see the second property described above) and they contain unique pieces of information that characterize the similarity between the two spatial distributions. Formally, the similarity between spatial distributions A and B can be represented by a 2-tuple or vector in a two-dimensional similarity space:

$$Sim(A, B) = Sim(B, A) = \langle NISD(A, B), NISD(B, A) \rangle \tag{5.14}$$

In our approach, the two-dimensional similarity space is a bounded. For example, $Sim(A, B) \in [0, MaxND^{(2)}] \times [0, MaxND^{(2)}]$ whether weighted mean or weighted square mean is used as the summarizing function. The degree of similarity between set A and B can be quantified by the length of vector $Sim(A, B)$, which is mathematically called norms of vectors in a vector space (denoted as $\|x\|$). The degree of similarity is in proportion to the length of the vector.

Without loss of generality, we use norms in L^p spaces (also referred to as L^p-norm) to define the length of similarity vectors (degrees of similarity) in a two-dimensional vector space:

$$\|Sim(A, B)\|_p = (|NISD(A, B)|^p + |NISD(B, A)|^p)^{1/p} \tag{5.15}$$

In the following we focus on two commonly used norms to illustrate how similarity between spatial distributions is calculated (the two norms are also shown in Fig. 5.25).

When $p = 2$, Equ. 5.15 actually gives a Euclidean length of the vector:

$$\|Sim(A, B)\|_2 = \sqrt{NISD(A, B)^2 + NISD(B, A)^2} \tag{5.16}$$

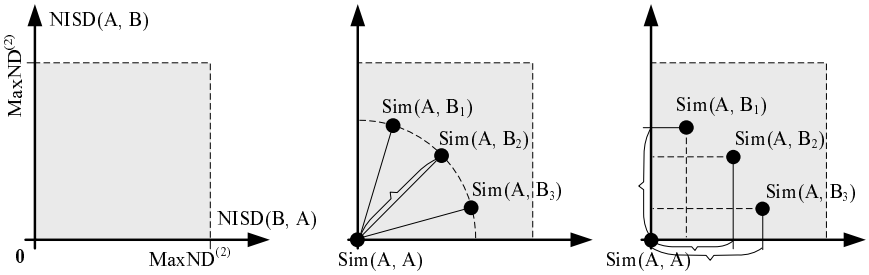


Figure 5.25 Similarity space (a) and examples of similarity measures based on L^2-norm (b) and L^∞-norm (c)

This equation defines a similarity measure of spatial distribution based on L^2-norm (Fig. 5.25b). The figure presents similarity comparisons between sets of locations A and B_1, B_2, and B_3 respectively. The L^2-norm similarity measure indicates that $\|Sim(A, B_1)\| = \|Sim(A, B_2)\| = \|Sim(A, B_3)\|$, which means that these three sets B_1, B_2, and B_3 are to the same degree similar to A in terms of spatial distribution.

When $p \to \infty$, the length of a vector is given by the so-called uniform norm:

$$\|Sim(A, B)\|_\infty = \max\{|NISD(A, B)|, |NISD(B, A)|\} \qquad (5.17)$$

This equation defines a similarity measure of spatial distribution based on L^∞-norm (Fig. 5.25c). In the figure all concerned location sets are the same to those used in in Fig. 5.25b. The L^∞-norm similarity measure however gives a slightly different result, that is, $\|Sim(A, B_1)\| = \|Sim(A, B_3)\| > \|Sim(A, B2)\|$. This means that B_1 and B_3 are to the same degree similar to A, while B_2 is more similar to A than are B_1 and B_3. Note that the equality such as $\|Sim(A, B_1)\| = \|Sim(A, B_3)\|$ does not necessarily follow that B_1 is identical to B_3.

Other values of p in the L^p-norm are also possible, and different values give rise to different similarity measures. For example, when $p = 1$, Equ. 5.15 gives a so-called *Manhattan distance*. Whatever choice of norms should be intuitively acceptable for a given application domain. Also note that no matter which definition is used, $\|Sim(A, B)\|_p = 0$ if and only if the similarity vector is a zero vector (e.g. $Sim(A, A)$ in Fig. 5.25). In this case, two sets of locations are identical (the fourth property of *NISD*).

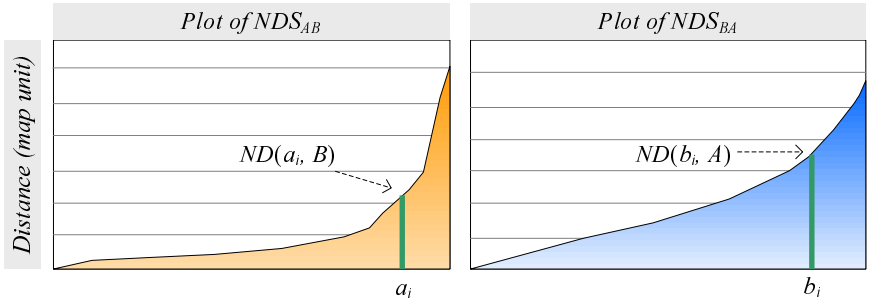


Figure 5.26 Plotting of NDS_{AB} and NDS_{BA} (A could be an initial distribution of (building) locations while B could be the target distribution of locations): vertical axis represents nearest distance from a point in one set to the other set; horizontal axis is ordered by distance values

Visualization and analysis of the nearest distances

The similarity measures defined above are summaries of the nearest distances in *NDS*. Apart from this, the similarity between two spatial distributions can be more intuitively visualized and analyzed based on the values in NDS_{AB} and NDS_{BA} respectively. This is illustrated in Fig. 5.26, where every bar (e.g. a_i and b_i) in the plot represents a nearest distance from a point in one set to the other set. If the two sets are similar in terms of spatial distribution, there are more smaller distance values and intuitively the areas formed in the plots become smaller; vice versa.

On the other hand, standard deviation of values in *NDS* can be used to further describe the variance between two spatial distributions. Usually, the smaller the standard deviation, the better the target distribution resembles the initial distribution. We will demonstrates the use of plot and variance by applying the presented approach to real topographic and artificial datasets in implementation (Section 7.6.2).

5.6.2 *Expressive power demonstrated*

It is difficult to understand why the similarity measures described in Section 5.6.1 is adequate for map generalization just from the way the approach is presented. This section gives several examples that explain how the measures are useful in measuring the similarity between spatial distributions. In the examples, we always refer to the initial (source) dataset as location set A and the target (generalized) dataset as B. Target datasets usually contain fewer objects than initial datasets.

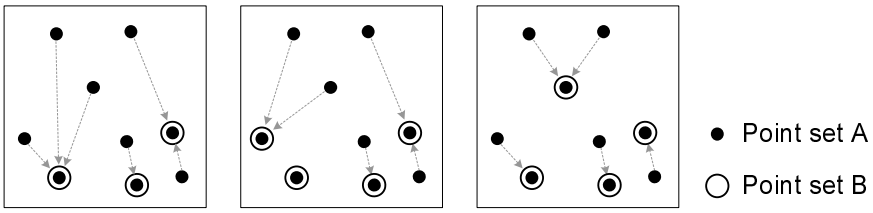


Figure 5.27 Three different selections from the initial points; arrows represent nearest distances from initial locations to target ones

Example 1

In the first example, the purpose is to compare the similarity between an initial distribution of locations (A) and three target distributions (B) that were generalized from A. The generalization was only based on selection of objects in the initial dataset (Fig. 5.27). As a result, it is apparent that $NISD(B, A) = 0$ and the similarity between the spatial distributions depends solely on $NISD(A, B)$. All the $ND(a_i, B)$ are displayed as dashed arrows, the total length of which indicates the magnitude of $NISD(A, B)$.

It is obvious that with the increase of selection ratio, $NISD(A, B)$ becomes smaller. For instance, in Fig. 5.27(b) and (c) one more object is selected than in Fig. 5.27(a). We can see from the arrows that $NISD(A, B)$ of both selection 2 and 3 are smaller than that of selection 1. A second observation is that given the same selection ratio, the similarity measures are able to distinguish between different selection schemes. That is, the more the target distribution is similar to the initial distribution, the smaller $NISD(A, B)$ is. Take selection 2 and 3 in Fig. 5.27 for example, the arrows in the figure indicate $NISD(A, B)$ for selection 3 is smaller than that of selection 2. This result is consistent with our perception.

Example 2

Similarity between spatial distributions that are of different map scales requires that the target distribution with fewer objects approximates the initial distribution as much as possible. In the previous example, only 1-to-1 corresponding relationship is concerned. In the second example, we will consider n-to-1 and n-to-m relationships that are common in map generalization. To better approximate some initial objects, a target object should seek for a better representative position to its surrounding initial objects

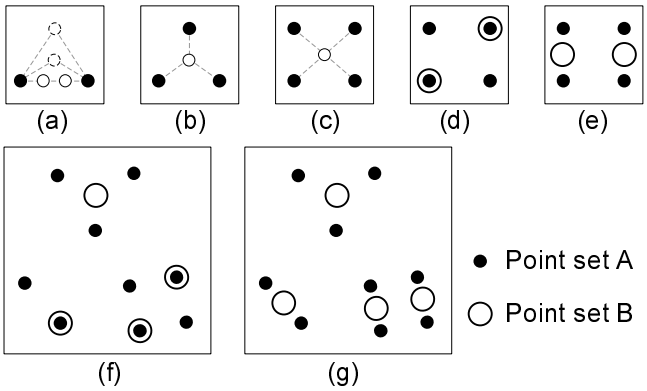


Figure 5.28 Expressive power of using F_1 and F_2 for calculating *NISD*: (a)–(e) are examples of n-to-1 relationship for which the optimal representative positions are determined; (f) and (g) are two other generalizations of the initial data in Fig. 5.27

so that the spatial distribution of all objects can be better preserved. This assumption is used in the typification algorithm based on Kohonen nets [172, 173].

Typification usually yields a n-to-m relationship between generalized and initial objects at a global level, while modeling such a relationship as a series of 1:n relationships gives more details of the relationship at a finer level [34]. This idea is also reflected in many typification algorithms [e.g. 41, 172, 173], where n-to-1 relationship is explicitly modeled by Delaunay triangulation. In our second example, we decompose the n-to-m relationship between objects in two datasets into several n-to-1 relationships to show the expressive power of the similarity measures.

In this example (Fig. 5.28), we show that the specific summarizing functions described in Equ. 5.13, namely weighted mean (F_1) and weighted square mean (F_2), have different expressive power in describing the similarity between spatial distributions. To begin with, Fig. 5.28(a)–(c) demonstrates the use of one target object to represent n initial objects. In Fig. 5.28(a), the representative object that lies in the segment connecting the two initial objects is considered better than in other positions, as positions on the connecting segment minimizes F_1. Nevertheless, F_1 is not adequate for distinguishing between different positions along the connecting segment. This is because whatever position the target object is on the segment, the value returned by F_1 always equals to the length of the segment. As such, in Fig. 5.28(d)

and (e) where two target objects are used to represent four initial objects, the two solutions are regarded to be the same by F_1.

F_2 is in this regard more expressive than F_1. Evidence can be drawn from the following analysis. If we view the problem of using one target object to represent n initial objects as an optimization problem, F_2 can be regarded as a function that can be optimized (minimized); when F_2 reaches its minimum, the optimal position for the representative target object can be drawn. The idea is mathematically demonstrated in the following.

Let x_i, y_i be the coordinate pair of $a_i \in A$, x, y be the unknown position of the target representative object to be optimized. We simplify the demonstration by letting all initial objects in A be equally weighted; then:

$$F_2(NDS_{AB}) = \sum_{i=0}^{n} \left((x - x_i)^2 + (y - y_i)^2 \right) \rightarrow \textit{Min}!$$

The unknown position x, y, for which the minimum of F_2 is approached, can be determined by the partial derivations of F_2 with respect to x and y and solving the following equations:

$$\frac{\partial F_2}{\partial x} = \sum (x - x_i) = 0 \quad \Rightarrow \quad x = \sum x_i / n$$
$$\frac{\partial F_2}{\partial y} = \sum (y - y_i) = 0 \quad \Rightarrow \quad y = \sum y_i / n$$

This demonstration indicates that the optimal position that the target object best approximate the n initial objects (equally weighted) is the geometric mean of the n initial locations. This is determined by *NISD* when F_2 is minimized. Hence, there was only one optimal position (geometric mean) for the situations presented in Fig. 5.28(a)–(c). Likewise, the solution in Fig. 5.28(e) is considered by F_2 to be more similar to the initial distribution than the solution in Fig. 5.28(d).

To summarize, several observations can be made. First, the two summarizing functions F_1 and F_2 were capable of identifying similar spatial distributions and distinguish dissimilar ones in the above examples. Second, the similarity measures (dominated by $\textit{NISD}(A, B)$) based on F_2 is theoretically more powerful than those based on F_1. $\textit{NISD}(A, B)$ on top of F_2 calculated for Fig. 5.28(g) is much smaller than that for Fig. 5.28(f), since in the former case F_2 reaches its minimum. It is interesting that this result is consistent with generalization requirements on preserving spatial distributions.

5.6.3 *Applicability to quality evaluation*

As demonstrated in Section 5.6.2, the proposed similarity measures of spatial distributions are expressive enough for quality evaluation of generalized data. When applying these measures to quality evaluation, one should consider the following side notes. First, the proposed similarity measures can be applied to areal features using their centroids. Second, if one wants to convert the similarity measurements into degrees of similarity, or constraint violations (satisfactions), a user-defined threshold can be used to normalize $\|Sim(A, B)\|$. This threshold should not exceed the theoretical bound *MaxND* or maybe even smaller. Furthermore, it is required to formalize the acceptable range in which spatial distributions should change at scale transitions, so that acceptable solutions can be qualified based on their degrees of similarity. A final note is that, the similarity measures are undefined if the target dataset (B) contains no object (non-empty set assumption violated). In such cases, set B is regarded to be very much dissimilar to set A unless A is also an empty set. The proposed similarity measures are validated using both real topographic and artificial datasets in Section 7.6.2.

6 Design of data matching and evaluation

6.1 Introduction

As described in the conceptual framework (Chapter 3), data matching is necessary for preservation constraints where target objects have to be evaluated with respect to initial objects, in which similarity is a key concept for matching data objects at different map scales.

6.1.1 *Motivation and general problems of data matching*

Geospatial data are usually collected for the same geographic areas from different sources and/or at different scales, and for different purposes. To make best use of different data sources, e.g., to carry out advanced spatial analysis based on different abstraction levels [50, 200], matching between datasets is needed. Meanwhile, to fulfill the increasing and diverse demand of spatial data at various resolutions and scales, detailed spatial databases are being built or under construction in many countries, from which smaller scale representations are derived via generalization. However, since fully automated generalization is currently not yet available, multiple representation databases (MRDBs) became a compromise [83, 170]. That is, spatial data of different levels of detail are stored separately and updates are propagated across scales. In the process data matching is key to establishing links between corresponding objects for the maintenance [102]. Additionally, to automatically assess the quality of generalized objects with respect to initial ones, links between corresponding objects are also required [192].

Matching spatial objects from two heterogeneous datasets is a complex decision process. To decide which pairs of objects match or are similar, we need different similarity measures and complex reasoning. Two fundamental problems arise. First, what are the key criteria (or variables) that help determine the matching. Second, how can we make a decision based on the multiple criteria.

6.1.2 *Approaches to data matching*

Previous work has been dedicated to the development of new similarity measures. In general, those measures can be divided into geometric, semantic and contextual measures. For instance, Beeri et al. [24] developed spatial join algorithms that match points only using their locations. To match more complex objects (polygons and networks), other geometric information such as angle, shape, and topological properties are also useful [205, 212]. Some other matching approaches also compare the semantics of objects, especially names [153], provided that the attribute was collected for the datasets.

A remarkable approach, proposed by Samal et al. [169], measures the contextual similarity between two buildings. The context (i.e. surrounding landmarks) of an object is captured in a proximity graphs, and the contextual similarity is calculated between two graphs using displacement vectors. In view of this, Kim et al. [103] represent context (also landmarks) by a triangulation structure, where the contextual similarity is measured based on areas and perimeters of the triangles organized around the building. This method is more reliable in case of large discrepancies existing between matching datasets. A limitation is that the matching of landmarks relies entirely on names, which is less applicable since names are not always available in topographic data. Note that, to use context one has to either match the context *a priori*, as in [169], or choose a unique context to which both datasets refers.

On the other hand, combining various matching criteria into a decision is still a challenge. Approaches based on a single criterion [e.g. 103] are free from this issue. However, single source of information does not provide enough evidence for a reliable decision. Therefore, we claim that data matching should combine multiple sources of information as evidence to improve the matching.

In this chapter, we present a generic approach and an *ad-hoc* approach to data matching. The generic approach attempts to provide a framework in which the corresponding relationship between spatial objects of different LODs can be established for various feature classes in a generic way (Section 6.2). The *ad-hoc* approach is specific to the matching of building patterns and alignments (Section 6.3) for automated evaluation of the patterns. After that, approaches to automatically evaluate the quality of generalized buildings in terms of building alignments and spatial distribution are described in Sections 6.4 and 6.5.

OWN PUBLICATIONS

This chapter is partly based on the following publications:

- X. Zhang, T. Ai, and J. Stoter. The evaluation of spatial distribution density in map generalization. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Congress 2008)*, volume XXXVII. Part B2., pages 181–188, Beijing, 2008.

- X. Zhang, J. Stoter, T. Ai, and M.-J. Kraak. Formalization and data enrichment for automated evaluation of building pattern preservation. In *Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science (SDH2010)*, volume XXXVIII, Part 2, pages 267–272, 2010 (paper selected as Chapter 9 in *Advances in Spatial Data Handling and GIS*, Lecture Notes in Geoinformation and Cartography, to appear in 2012).

- X. Zhang, X. Zhao, M. Molenaar, J. Stoter, M.-J. Kraak, T. Ai. Pattern classification approaches to matching building polygons at multiple scales. In *(ISPRS Congress 2012)*, Melbourne, 2012.

- X. Zhang, J. Stoter, T. Ai, M.-J. Kraak, and M. Molenaar. Automated evaluation of building alignments in generalised data. *Manuscript submitted for review to an academic journal*.

6.2 DATA MATCHING USING PATTERN CLASSIFICATION

In this section, we present a generic approach to data matching. This generic approach aims to tackle the multivariate decision problem, i.e., how to combine different criteria or similarity measures to make final decisions (see Section 6.1.1). A straightforward approach to this problem is weighted average which is commonly used in data matching approaches [e.g. 169, 212] and consists of two steps: (i) normalizing measured values, and (ii) assigning weights to different measures. Clearly, both steps can be problematic. For one thing, normalization factors may not always be available. For another, manual weighting is usually subjective; even experts may fail to assign appropriate weights simply because weight assignment is counter-intuitive. Additionally, as data matching is essentially an uncertain process (see also Section 4.1.3), crisp decisions would inevitably reduce the matching performance.

To address these issues, we fit the data matching into a pattern classification framework, which are particularly effective in solving multivariate decision problems. One advantage is that the model parameters can be learned from available data, and the subjective weighting can hence be avoided. Moreover, some of the classification methods (e.g. probabilistic ones) can handle uncertainties, which may help to improve decisions in ambiguous situations. Although we use building features as a running example to explain the proposed generic matching approach, this approach should also be applicable to the matching of general spatial objects such as linear, areal, and network features.

In the following, Section 6.2.1 formalizes the data matching into a pattern classification problem. Then basic geometric criteria and supervised classifiers are introduced in Sections 6.2.2 and 6.2.3. An extension is presented in Section 6.2.4 which integrates soft classification and domain knowledge to improve the matching. The pattern classification approaches to data matching were implemented and validated in Section 7.4.

6.2.1 *Problem formalization*

Data matching aims to find all possible corresponding pairs of entities (individuals or groups) from two datasets based on some criteria. Each criterion compares a specific characteristic (e.g. shape or orientation) between a pair of entities and yields a measured value. Based on the measured values a decision can be made as to whether or not the two entities correspond to each other. In the following, we formalize this problem as a pattern classification problem.

First, let $r_{ij} = (d_i, g_j) \in D \times G$ be a relation between a pair of entities, $d_i \in D$ and $g_j \in G$, where D and G denote detailed (initial) and generalized (target) datasets, respectively. Hence, all possible relations between the initial and target datasets can be described by the Cartesian product of the two datasets ($D \times G$). Normally, all corresponding (matching) relationships between the two datasets can be viewed as a subset of $D \times G$. In this respect, any relation $r_{ij} \in D \times G$ can be categorized as either 'Matched' or 'UnMatched'.

In this respect, data matching can be viewed as a two-category pattern classification problem with category C = {'Matched', 'UnMatched'}. The characteristics of the matching candidates and similarity measures used to help decide the matching can be regarded, in pattern classification terms, as a feature vector $(f|f_1, \ldots, f_n)$ and feature extractors, respectively. In

other words, r_{ij} can be classified into a category $c_k \in C$, depending on the extracted feature vector. At this point, a pattern point can be denoted as $(r_{ij}; f|f_1, \ldots, f_n)$, or shortly $(r_{ij}; f)$.

Formally, there exists an unknown function $g : D \times G \rightarrow C$ that maps an input pattern $(r_{ij}; f)$ to a category label c_k. However, since such an ideal function is not available for real applications, most classification approaches learn from training patterns $TP = \{(f_1, c_1), \ldots, (f_n, c_n)\}$ and produce a function $h : D \times G \rightarrow C$ that approximate g as closely as possible. Such classification approaches are termed supervised classifiers.

In the following, we examine several supervised classifiers on their capability of learning and classification (i.e. prediction) by applying them to the matching of building polygons. Before that, some basic measures are firstly introduced in the next section, based on which the feature vector can be extracted.

6.2.2 *Selected criteria*

Several measures have been proposed in previous work for the matching problem. Here, instead of introducing more complex measures, we select four commonly used measures that are especially suitable for matching building polygons, namely similarity in position, size, shape and orientation. Besides, according to some pattern classification approaches (e.g. Naive Bayes classifier), independent features are preferred over dependent ones.

Similarity of positions

Position similarity is crucial since different representations of a same building at neighboring map scales usually locate closely to each other. A large number of distance measures are pertinent to data matching problems (see e.g. Section 5.2.1). Nevertheless, those measures all combine aspects of other characteristics (e.g. shape), and therefore they are not independent measures. Here, Euclidian distance between building centroids is selected to measure position similarity.

Similarity of sizes

Size similarity also contributes to the matching of buildings at different scales. Here, we define size similarity based on the following size ratio:

$$SizeSim(d_i, g_j) = \frac{Area(d_i)}{Area(g_j)} \tag{6.1}$$

This size similarity measure is interpreted as follows: when $SizeSim(\cdot)$ more approaches to 1, the two buildings are more similar in size.

Similarity of shapes

Shape index (Equ. 5.1) is used here to compare the similarity of building shapes. The choice is also justified in Section 5.2.3. Shape similarity between a pair of buildings (matching candidates) is based on the ratio of shape index:

$$ShapeSim(d_i, g_j) = \frac{ShapeIndex(d_i)}{ShapeIndex(g_j)} \tag{6.2}$$

Similarity of orientations

Orientation of building polygons is measured using adapted version of wall statistical weighting (see Section 5.2.2). Instead of measuring the similarity of building orientation, we measure orientation deviation between two buildings. The smaller the deviation is, the more similar two buildings are; the two become the most dissimilar when the deviation reaches its maximum ($\pi/2$). The deviation between two building orientations is defined as:

$$\begin{cases} Dev(d_i, g_j) = |WSW(d_i) - WSW(g_j)| \\ Dev(d_i, g_j) = \pi - Dev(d_i, g_j), & \text{if } Dev(d_i, g_j) > \pi/2 \end{cases} \tag{6.3}$$

where $WSW(\cdot)$ represents the adapted wall statistical weighting approach which returns a measured orientation and its confidence value. Note that in Equ. 6.3 $Dev(\cdot)$ is fitted into $[0, \pi/2]$.

However, this deviation cannot be used directly as the final orientation difference because of the following reason. That is, many more elongated buildings (having strong major wall directions) represented in the more detailed dataset (e.g. the initial building in Fig. 6.1), are represented by less elongated shapes such as squares in the generalized dataset (e.g. the target building in Fig. 6.1). In such cases, the wall statistical weighting would make a random choice as to which one of the two wall directions is the major direction (as illustrated in Fig. 6.1). Consequently, Equ. 6.3 could give a result of $\pi/2$ for the two buildings in Fig. 6.1, indicating that they are extremely different in orientation. This is however not the case, as the two represent the same real-world object.

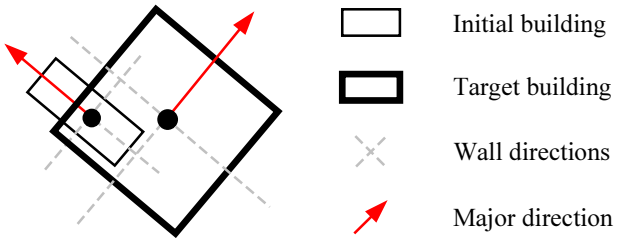


Figure 6.1 Initial and target buildings with strong and weak major directions

To better account for this, we use a confidence threshold T_C to distinguish between strong and weak major wall directions. If the measured confidence value is less than T_C the building is regarded to have a weak major wall direction, and vice versa. Furthermore, if at least one of the matching candidates has a weak major wall direction, the orientation difference between the two should not exceed $\pi/4$. Only if both buildings have strong major directions, the orientation difference is determined the absolute deviation calculated by Equ. 6.3. According to this, a new equation for orientation difference is formulated:

$$OriDiff(d_i, g_j) = \begin{cases} \pi/2 - Dev(d_i, g_j), & \text{if } Dev(d_i, g_j) > \pi/4 \\ & \wedge(Con(d_i) \leq T_C \\ & \vee Con(g_j) \leq T_C) \\ Dev(d_i, g_j), & \text{otherwise} \end{cases} \tag{6.4}$$

where $Con(\cdot)$ is the confidence value coming along with the wall statistical weighting orientation. In this study, a threshold value ($T_C = 0.55$) was empirically determined from training data.

To sum up, the pattern classification approaches used in the subsequent sections solely rely on these four similarity measures. Note also that all these measures do not need to be normalized.

6.2.3 *Learning and classification approaches*

We give a brief introduction to the fundamentals of several supervised classifiers, including decision trees (e.g. classification and regression tree and C4.5 algorithms), probabilistic models (e.g. Naive Bayes classifier) and Support Vector Machines (SVM). More detailed information and mathematical treatments can be found in Duda et al. [60].

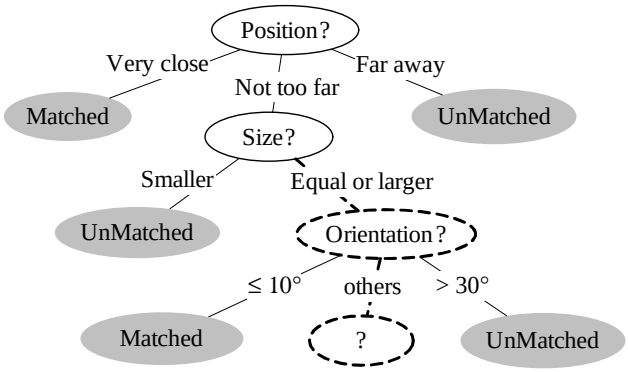


Figure 6.2 Illustration of a decision tree for the data matching problem

Decision trees

A decision tree is simply a sequence of questions organized in a (directed) tree, where the first question (root node) is at the top, connected consecutively by links to other questions (nodes) or a final decision (leaf node); Fig. 6.2 illustrates a decision tree for the building matching problem. In our application the tree is actually a binary tree (with 'yes' or 'no' answers to questions like if the distance between objects is less than X mm). One of the benefits of trees over many other classifiers is their interpretability [60]. That is, we can easily interpret the decision for any pattern as logical conjunctions (rules) when classifying new data. Besides, trees also enable rapid classification over a large amount of data. Next, we briefly describe two tree-growing methods that create trees from training data. They are Classification and Regression Trees, or shortly CART [39], and C4.5 algorithms [152].

In principle, tree-learning algorithms (e.g. CART and C4.5) progressively split the training data into subsets, and at each split the algorithms automatically choose among others an optimal feature (e.g. position or shape) that best separates the training data. To formally measure "best", various (entropy-based) impurity indices are defined for specific algorithms [60]. Ideally the splitting proceeds until the samples in each subset have the same category. But more commonly, rules are used to stop splitting at certain point; or alternatively, the splitting continues until the descendent nodes are completely pure and then pruning is used to merge sibling nodes to obtain a certain degree of purity. In either case, an imperfect decision node is created.

Technically, CART and C4.5 differ in their choices for stopping rules and for pruning methods. In our implementation, we firstly used a rule for CART

to stop splitting when the majority class rate reaches MR(%); whereas no stopping rule was set for C4.5. For clarity, CART with different versions is denoted CART^{MR}, with $\text{CART}^{\star}$ indicating no stopping rule is applied. Then, unlike the impurity-based pruning employed in CART, C4.5 uses information derived from the statistical significance of splits [60, p. 411]. Besides, C4.5 employs so-called C4.5Rules to delete redundant rules that are generated in the learning process. This in turn reduces the size of generated trees.

Naive Bayes classifier

The idea behind Bayes classification is the class (C) of any input pattern point is conditionally dependent on a set of features $(F_1, \ldots, F_n)$. Naive Bayes classifier is the simplest probabilistic classifier as it assumes that the features used are conditionally independent given the class. Formally, the underlying probability model is a conditional model:

$$P(C|F_1, \ldots, F_n)$$

The expression means the probability of an object being classified as C given $F_1, \ldots, F_n$. Using Bayes' theorem, the above expression can be rewritten as:

$$P(C|F_1, \ldots, F_n) = \frac{P(F_1, \ldots, F_n|C)P(C)}{P(F_1, \ldots, F_n)}$$

Under the naive assumption of conditional independence, the first term of the numerator in the above equation can be reformulated as:

$$\begin{aligned} P(F_1, \ldots, F_n|C) &= P(C)P(F_1|C)P(F_2|C)\cdots P(F_n|C) \\ &= P(C)\prod_{i=1}^{n} P(F_i|C) \end{aligned}$$

Thereafter, the probability model for Naive Bayes classifier can be defined as:

$$P(C|F_1, \ldots, F_n) = \frac{1}{Z}P(C)\prod_{i=1}^{n} P(F_i|C)$$

where Z is a scaling factor that depends only on $F_1, \ldots, F_n$ and hence it is a constant. Naive Bayes model classifies a given pattern point $(f_1, \ldots, f_n)$ depending on the class C for which its posterior probability reaches maximum, hence Z can be ignored in the classification decision. Formally, Naive

Bayes classification is given below:

$$\text{classify}(f_1, \ldots, f_n) = \arg\max_{c_k} P(C = c_k) \prod_{i=1}^{n} P(F_i = f_i | C = c_k) \qquad (6.5)$$

Such a probability model is much easier for machine learning due to its independence assumption, which is practically fairly useful even though the features are not independent [60]. In particular, the class-conditional distribution $P(F_i|C)$ can be estimated independently for each feature and for each class as a one dimensional distribution. This in turn helps to alleviate problems such as the curse of dimensionality [60]. That is, the required amount of training samples grows exponentially with the number of features, which would be otherwise intractable for those dependent feature models. In our implementation, we used a normal kernel to estimate each class-conditional density distribution from training data.

Support vector machines

Support Vector Machines (SVM) are based on the idea of finding a decision plane that distinguishes between two classes. It relies fundamentally on the preprocessing of data: transforming pattern points into a higher (maybe infinite) dimensional metric space with a nonlinear mapping (kernel). With an appropriate mapping to a sufficiently high dimension, patterns of two classes can always be separated by a hyperplane [60].

In our implementation, we used a radial basis function (RBF) kernel. With the RBF kernel, the best combination of $C > 0$ (penalty parameter of classification error) and the kernel parameter γ should be found for an optimal hyperplane. Instead of trying out different combinations manually, we used LIBSVM[1] package which automatically induce parameters from training data using a grid search strategy with cross-validation (we used a 10-folds cross-validation). For clarity, we use $\text{SVM}^{C,\gamma}$ to denote SVM with different parameter values.

6.2.4 *Incorporating domain knowledge*

The generalization knowledge can be used to improve the classification results. Note that the knowledge can only be integrated with classifiers that can handle uncertainties (e.g. Naive Bayes). The following two rules only applies to individual objects but not to groups:

Rule 1. Any target object should link to at least one initial object;

[1] LIBSVM: http://www.csie.ntu.edu.tw/ cjlin/libsvm/

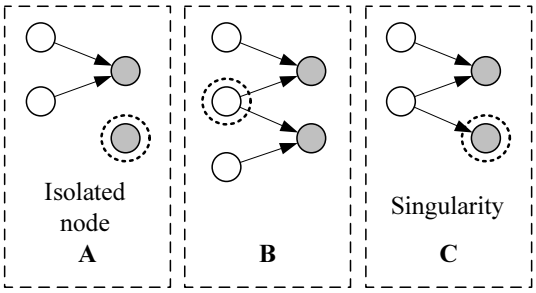


Figure 6.3 Problems and ambiguous situations between linked initial and target objects

Rule 2. Any initial object should link to one most probable target object.

We further distinguish between three ambiguous situations (Fig. 6.3) where the above-mentioned rules are violated:

A. No initial objects is linked to the target building (this target building is called *isolated node*)

B. In a cluster (group of objects connected by the links), an initial building has more than one link to target buildings;

C. Similar to case B, an initial building links to more than one target buildings, but differently, one of the target buildings is linked only by the initial building, creating a *singularity*.

These situations can be improved by the following step-by-step refinement:

1. Link every isolated node (A in Fig. 6.3) with the most probable candidate;

2. For each cluster, if there is no singularity (B in Fig. 6.3), select the most probable link from initial buildings and remove less probable ones;

3. Otherwise, for each identified singularity s_i, cut all links from initial building d_i except for the link between d_i and s_i and update the cluster;

4. Repeat steps 2 and 3 until none of the above three situations can be found.

The domain knowledge can be incorporated into a classification process as a post-processing to refine matched pairs of objects with uncertainty values. We will show how such refinement improves the matching accuracy in Section 7.4.

Figure 6.4 Examples of changes of building patterns at scale transitions from (a) 1:10k to 1:50k and (b)–(d) 1:50k to 1:100k (Data source: Kadaster, NL)

6.3 Matching of building patterns

We have described the typology (Fig. 5.6) and automatic recognition of building patterns (clusters) and alignments in Sections 5.3 and 5.4. In this section, we present *ad-hoc* approaches to matching building patterns and alignments in particular. This is motivated by the requirement to automatically evaluate the quality of generalized building alignments with respect to the initial alignments. The *ad-hoc* approaches are more specific to building patterns, and therefore they can be implemented in a much quicker and easier way than the generic matching approach described in Section 6.2.

First of all, Section 6.3.1 observes possible changes of building patterns and alignments at scale transitions, i.e., how the patterns were generalized. After that, *ad-hoc* matching methods are presented in Sections 6.3.2 and 6.3.3 to match building patterns and alignments, respectively.

6.3.1 *Changes of building patterns at scale transitions*

To get more insights into how building patterns change at scale transitions, I visually study and analyze topographic map series of Dutch Kadaster

Initial scale	Multiplicity	Target scale	Examples
Each pattern type	1-to-1	The same type	B in Fig. 6.4(a) and (c); A in Fig. 6.4(b)
Unstructured	1-to-1	Grid-like pattern	A in Fig. 6.4(c)
Unstructured	1-to-1	Alignment	B in Fig. 6.4(d)
Grid-like	1-to-1	Collinear alignment	
Alignment	n-to-1	Alignment	A in Fig. 6.4(d)
Alignment	n-to-1	Grid-like pattern	B in Fig. 6.4(d)
Alignment	n-to-1	Unstructured	
Nonlinear cluster	n-to-1	Nonlinear cluster	C in Fig. 6.4(d)
Align-along-road	1-to-1	Collinear/curvilinear	A in Fig. 6.4(b)

Table 6.1 Different ways in which building patterns are kept at scale transitions

at scales 1:10k, 1:50k, and 1:100k (see e.g. Fig. 6.4). Maps at scale 1:10k (supported with an object oriented database, called TOP10NL) is the most detailed topographic database, from which maps at scale 1:50k (supported with TOP50NL database) are interactively generalized. Maps at scale 1:100k (supported with TOP100NL database) are interactively generalized from 1:50k maps. By comparing building patterns at these scales, the following transitional knowledge was obtained.

In general, we observe three kinds of pattern transitions. First, some building patterns are diminished. For example, an initial alignment consisting of three buildings is diminished after two buildings have been removed, or the entire alignment is removed. This may be because in some situations respecting minimum distance is more important than keeping building alignments. Second, some building patterns are transformed into built-up areas. This is common in the Dutch case due to a constraint stating that areas of high building densities should be represented by built-up areas [see e.g. 193]. We also found that some built-up areas are based on the areas that are not fully surrounded by the road network but are based on extents of initial alignments of higher densities (A in Fig. 6.4(a)). Third, the building patterns are preserved.

There are many different ways in which building patterns can be preserved (as is summarized in Table 6.1). For example, members of a building alignment can change without changing the the overall alignment; an alignment can become part of another larger alignment or cluster [46]. As discussed

in Section 4.1.3, inter-scale relationships between building patterns and alignments can be complicated. For example, the n-to-1 relationship is common between different types of patterns. As an exception, the 0-to-1 (different from n-to-m) correspondence also exists for alignments, that is, when initially irregularly distributed buildings (non-alignment) become a regular alignment after generalization.

These changes of building patterns are mainly caused by two cartographic conflicts which in turn drive the generalization of individual buildings. They are densely populated buildings and road networks, which leads to the changes from one pattern type to another. For example, if building groups enclosed by roads are too dense, they are usually typified or aggregated into street blocks, leading for example to the change of a grid-like pattern into an alignment. If road networks are pruned or simplified, several patterns may form a larger pattern. Note that the multiplicity of is specified at the meso-group level rather than at the individual level (see Section 4.1.3).

6.3.2 *Automatic matching of building patterns*

The matching process consists of two sub-processes: geometric and semantic matching. The matching is based on the previously enriched information concerning building pattern descriptions and transition events. The descriptions include pattern type information, representational geometries and homogeneity properties (Section 5.3.2). When a generalized pattern is geometrically matched with an initial pattern, type information of the two patterns is checked with respect to Table 6.1. If the type information is consistent with the transition events, the matching pair is selected as a candidate for further matching; otherwise the two patterns are regarded to be not corresponding to each other. In the latter case, one can conclude that the initial pattern is not preserved during generalization. It is worth noting that the buildings to be evaluated are partitioned by road networks and each step of the automatic matching is restricted to buildings within a partition unit. The technical detail of the process is described as follows.

The geometric matching deals with the similarity between geometries of the patterns to be matched. In the matching of the same type patterns, the similarity can be measured by distances like nearest distance, *Hausdorff distance*, and *Fréchet distance* [12]. To match polygons (nonlinear clusters), contrast model for similarity measure developed by Tversky [202] is appropriate. In the cases where linear alignments are to be matched with nonlinear clusters, buffers of skeletons of the alignments should be used instead to improve

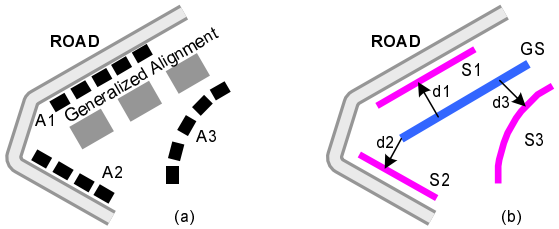


Figure 6.5 Automated matching of building patterns: (a) a matching situation and (b) geometric and semantic matching

the matching result.

The matching result obtained from geometric matching can be refined by semantic matching, especially when the result contains several candidates. When matching the patterns of the same type, the characteristics like pattern type, align angle, curvature and aligned road are crucial. For example, Fig. 6.5(a) describes a case where a generalized collinear alignment (the gray buildings) is (potentially) matched with two collinear patterns (A1 and A2) and a curvilinear one (A3), as they are equally close to each other (d1 = d2 = d3 in Fig. 6.5(b)). Such an ambiguous situation is caused because A1 was displaced to avoid the widened road symbol. To improve the matching, pattern type is used to filter out the curvilinear pattern A3, and then the angle of the skeletons (S1, S2, S3, and GS) is calculated. Since the angle between GS and S1 is the smallest, the method matches the generalized alignment to A1.

6.3.3 *Algorithm to match building alignments*

This section describes an algorithm to match corresponding alignments (including collinear and curvilinear ones). Since there is only one type of building patterns, the semantic aspect is not considered which considerably simplifies the design of the matching algorithm. The matching process is based on the geometries (skeletons) of alignments. In describing the algorithm, we refer to target alignments as sources and initial alignments as matching candidates. The matching starts from target alignment: it firstly searches for matching candidates in the initial dataset (some source alignment may have several candidates) and then filters out less possible ones.

In the following, we firstly present the detailed algorithm (Section on 'THE ALGORITHM') and then discuss the issue of partial matching (Section on 'PARTIAL MATCHING') and the confidence indicator used to address the issue (Section on 'CONFIDENCE INDICATOR').

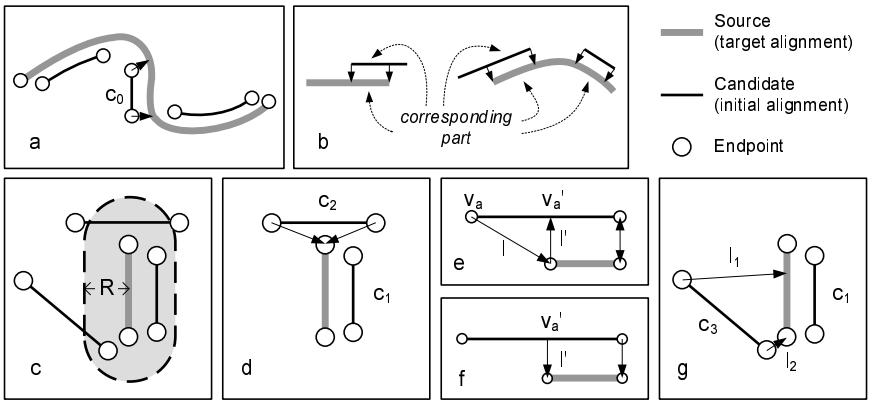


Figure 6.6 Matching of alignments: matching problems (a)–(b); searching and filtering rules (c)–(g)

The algorithm

We use position and orientation as filtering rules. That is, initial alignments that are more distant from the source and that are more different from the source in terms of orientation are unlikely to be matched and hence are filtered out. The orientation rule, however, is only appropriate for collinear alignments, since some curvilinear alignments have only local orientations but no definitive global orientation. Therefore, using alignment orientation can be misleading. For instance a collinear alignment (c_0 in Fig. 6.6(a)) is to be matched with part of a curvilinear one, or two curvilinear alignments are to be matched. In this regard, we claim that parallelism is a more appropriate concept to measure the similarity between corresponding alignments. The detailed matching process is described as follows.

To begin with, buffers with radius R are generated for each source alignment, and the initial alignments that intersect, touch or are inside the buffer are identified as candidates (Fig. 6.6(c)). For building alignments, R can be estimated from suburban or rural areas where buildings are largely displaced. Buildings in densely populated urban areas are conversely less displaced due to limited free spaces. In our experiment, we used $R = 0.5$ mm (map unit at 1:50k is used throughout this paper) to search for matching candidates at 1:10k.

Second, since the condition of being parallel is practically difficult to measure between arbitrary curves, we implemented it in a simplified way which

is specific to the characteristics of alignments. In the following, we use endpoints of alignments, as well as their potential corresponding vertices on their corresponding alignments, to determine whether or not the two are (partly) parallel. This vertex level matching also helps to identify the corresponding parts (Fig. 6.6(b)) by which the source and candidate actually correspond to each other (i.e. partial matching). The algorithm is described as follows.

For each source alignment:

1. start with two endpoints of a candidate alignment and search their potential corresponding vertices on the source based on distance; the closest vertices are mostly the corresponding ones; initially, anchor vertices for the candidate are its two endpoints and anchor vertices for the source are the identified corresponding vertices on the source alignment;

2. if the two endpoints correspond to a same vertex, which is also one of the endpoints, of the source alignment, the candidate (c_2 in Fig. 6.6(d)) is filtered out and the procedure jumps back to step 1 with another candidate;

3. otherwise, adjust anchor vertices for the candidate: if the corresponding vertex of any one endpoint (initial anchor vertex v_a) of the candidate is an endpoint of the source, and if this corresponding vertex links to a vertex (v'_a) other than v_a (Fig. 6.6(e)), this procedure adjusts the anchor vertex from v_a to v'_a as well as the relevant links (Fig. 6.6(f)). One exception is that when $|\mathit{Length}(l) - \mathit{Length}(l')| < \mathit{AdjustLength}$, where l and l' are the initial and adjusted link (Fig. 6.6(e)), and in such situations the procedure returns to the initial anchor vertex;

4. if the two links (after anchor vertices have been adjusted) satisfy: $|\mathit{Length}(l_1) - \mathit{Length}(l_2)| > \mathit{ParallelLength}$, the candidate alignment is regarded to be obviously unparallel to the source and is thus filtered out (e.g. c_3 in Fig. 6.6(g)).

The introduced parameter *AdjustLength* in step 3 is to allow for a larger degree of flexibility. That is, when the difference between the length of links before and after adjustment is subtle, it is preferred not to adjust the initial anchor vertex. As a result of this flexibility, the corresponding relationship in Figure 4c can be correctly identified. Step 2 and 4 also succeeded in filtering irrelevant candidates out (see Fig. 6.7(g) and (h)). *AdjustLength* and *ParallelLength* (the latter should be reasonably larger than the former)

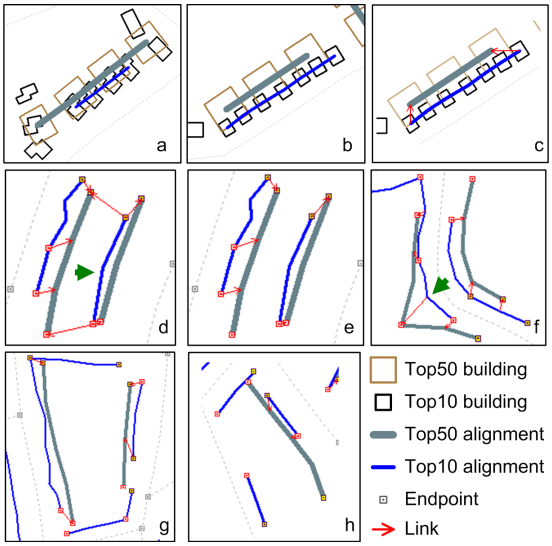


Figure 6.7 Matching of alignments: (a)–(b) possible changes of alignments at scale transitions; (c)–(h) examples of matching results (lines between two endpoints represents individual alignments; red links identify the corresponding parts)

were experimentally set to 0.16 mm and 0.6 mm, respectively.

A consistent checking is required to make sure the following assumption. That is, one initial alignment should not correspond to more than one target alignment in the way demonstrated in Fig. 6.7(d) (the one highlighted by green arrow). Such inconsistency could have occurred in densely populated areas if the buffer parameter R was too large. Sometimes, 1-to-n relationship is acceptable when it looks like the one in Fig. 6.7(f) (highlighted by green arrow). Thus to identify and remove the real inconsistency, the process checks initial alignments that links to more than one target alignment: for each such initial alignment, different target alignments correspond to this initial alignment on different parts; those target alignments, whose corresponding parts on the initial alignment overlap or are identical, are not consistent with this initial alignment, and the one that is the closest to this initial alignment is identified as a real correspondence (Fig. 6.7(e)).

Finally, the sources and their remaining candidates are considered as correspondences. 1-to-0, 1-to-1 and n-to-1 relationships can be observed between initial and target alignments (see Fig. 6.7). The matching algorithm was implemented and more results are presented and discussed in Section 7.5.3.

Partial matching

A visual inspection of the matching results (Section 7.5.3) shows that partial matching (matching between alignment parts) was rather common while the matching accuracy in our experiment was 100%. This means that partial matching was not a result of poor matching. In fact, the partial matching between alignments is inherent because of the following reasons. First, usually alignments were not explicitly treated as a whole but the composing objects were manipulated during the generalization, and hence some of the composing objects might be removed to meet readability constraints. As a result, some target alignments may contain fewer members (see e.g. Fig 6.7(b)) and others may as a whole be deleted. Second, an alignment may become part of another alignment at the target scale, meaning that other buildings that are not part of this alignment at the initial scale become its new parts (Fig 6.7(a)), or several alignments are merged into a longer alignment. More complicated situations could be combined from the two.

According to related map specifications (Table 6.2), every alignment is evaluated as a whole but only the parts are matched to one another. Such inevitable partial matching has meaningful impact on, and hence has to be dealt with in, the subsequent evaluation processes (see Section 6.4).

Confidence indicator

Generally, the partial matching mentioned above implies that the evaluation system is not always confident about its evaluation results. However, since the partial matching is unavoidable, we introduce a confidence indicator to quantify the degree of confidence of a system statement. This way intended users could be better informed of the reliability of the quality assessments. The confidence indicator is based on the degree of partial matching, or formally $MRatio(A_i) = MVNum(A_i)/VNum(A_i)$, denoting the ratio between the number of matched vertices (buildings) in the corresponding part and the total vertices in the alignment. Note however that the confidence indicator, i.e., *MRatio*, should be calculated differently depending on which subject in a corresponding relationship (i.e. initial alignment, target alignment, or both) the system intends to make statement about. To explain this idea, we use the following example (Fig. 6.8).

Two cases are shown in Fig. 6.8, in which some of the initial buildings are eliminated during the generalization on purpose to show the idea of matching ratio. Green links in the figure indicate the matched corresponding parts in the partial matching. In case 1, three buildings are selected to represent

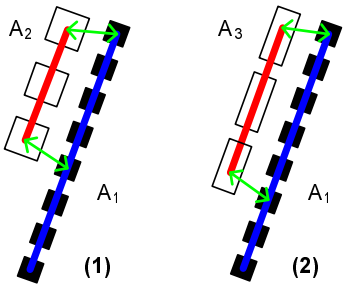


Figure 6.8 Partial matching and confidence indicator calculation (initial buildings are in black; target ones are in white)

the initial alignment. The three target buildings have been generalized using elimination and typification. In this case, $MRatio(A_1) = 5$ matched initial buildings / 8 total initial buildings = 0.625. In case 2, on the other hand, the top six initial buildings were aggregated into the three bigger buildings (every two into one). In this case, $MRatio(A_1) = 6$ matched initial buildings / 8 total initial buildings = 0.75. For both cases, the matching ratio for target alignments is the same, that is $MRatio(A_2) = MRatio(A_3) =$ 3 matched target buildings / 3 total target buildings = 1.

The calculation of *MRatio* can be different depending on the constraints to be evaluated. We will discuss the different calculations for the evaluation of building alignments in generalized maps in Section 6.4.

6.4 Evaluation of building alignments

After the recognition and matching of building alignments of different LODs, the next step in the conceptual framework (Sections 3.1 and 3.5) is to evaluate the quality of building alignments at scale transitions. To begin with, we analyze the constraints related to the alignment structure which defines how the automated evaluation should be carried out (Section 6.4.1). After that, Section 6.4.2 describes how to calculate constraint violations (constraint evaluation). Results obtained from the presented evaluation method are reported in Section 7.5.

6.4.1 *Analysis of constraints on alignments*

The structured constraints in Table 6.2 provide a starting point for the formalization and automated evaluation of the alignment structure. In this table, three specific aspects of building alignments that need to be preserved are identified. These are the existence (i.e. initial entities should be kept in the target dataset), orientation of alignment, and spatial distribution of composing buildings. To summarize, constraint 3 is defined specifically

ID	Geometry type	Kind of group	Kind of objects of the initial data composing the group	Constrained property	Condition to be respected
1	point areal	Alignments		Alignment	Alignment should be kept
2	point areal	Alignments		Alignment orientation	Target orientation should be similar to initial orientation
3	areal	Building alignment	Buildings aligned	Spatial distribution	Target distribution should be similar to initial distribution

Table 6.2 Harmonized constraints on alignments in the EuroSDR project [192]

for building alignments and the other two for alignments in general. The measures corresponding to the constraints are introduced in Section 5.4.5. Note that, since building alignments are groups of regular distributed buildings, homogeneity measure is used to quantify their regularities (we also discussed this in Section 5.5). Formal expressions of the three constraints are given previously (see C6–C8 in Section 4.3).

In Section 3.2.3 we have distinguished between entity-level and property-/relation-level preservation. According to the distinction, the constraints in Table 6.2 can be analyzed as follows (see Fig. 6.9). Constraint 1 is an entity-level preservation since the constrained property is the subject (kind of group) of the constraint; whereas constraint 2 and 3 are property-/relation-level preservation. Note also that the subject of constraint 1 and 2 is at the group level; whereas the subject of constraint 3 is the group members and the constrained property is the collective relationship among the the composing objects.

6.4.2 *Constraint evaluation*

We will now describe ways to evaluate the three aspects of building alignments, namely the existence of alignments, alignment orientation and the spatial distribution of composing buildings. The content of each constraint is organized as follows: we first describe how to evaluate the constraint for individual alignments and then how to calculate the degree to which the

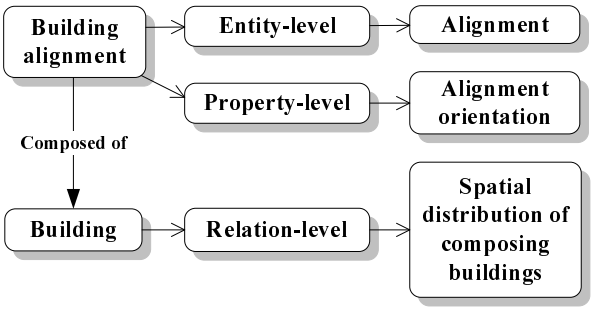


Figure 6.9 Analysis of constraints on alignments in terms of preservation types

system is confident about the result (confidence indicator). Finally, the aggregation of individual assessments is presented to summarize the quality about the whole feature class concerning each of the constraints.

Before proceeding to the evaluation processes, distinctions should be made between initial alignments $\{A_i|i = 1, \ldots, n\}$, matched initial alignments, target alignments $\{A_t|t = 1, \ldots, m\}$, matched target alignments and the correspondence relationship (Fig. 6.10). Unmatched alignments are complement of matched alignments in a dataset. In particular, unmatched initial alignments are the ones that have been deleted during generalization, and hence have no corresponding alignments at the target scale (1-to-0 relationship). Unmatched target alignments are the ones that have emerged from irregularly distributed groups of buildings, and hence have no corresponding initial alignments at the initial scale (0-to-1 relationship). Since currently no constraints address the newly emerged alignments, such alignments are excluded from our evaluation and are not counted in the aggregation phase too.

Existence of alignments

First of all, we use CV (Section 3.3.3) to denote the degree of constraint violation ($CV \in [0, 1]$, with $CV = 1$ meaning the constraint is violated and 0 meaning the constraint is completely satisfied). The constraint 'Alignment should be kept' protects initial alignments from being deleted, so every initial alignment should be checked to see if it still exists at the target scale. This is done by checking corresponding relationships obtained from the matching process (Section 6.3.3). The evaluation results in a Boolean output for each alignment: either the alignment is kept or it is diminished. If an initial alignment (A_i) was removed the existence constraint is violated ($CV_{Existence}(A_i) = 1$), or it is completely satisfied ($CV_{Existence}(A_i) = 0$) for this alignment. In the cases that initial alignments were removed (the

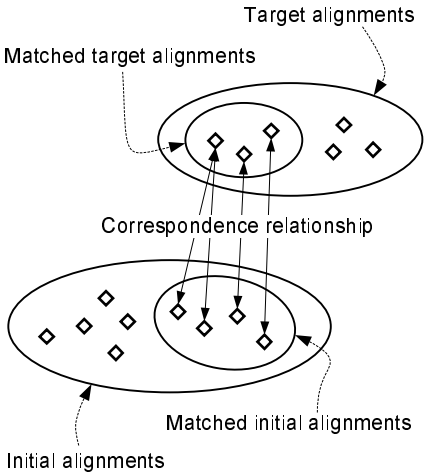


Figure 6.10 Venn diagram representing sets of alignments (circles), individual alignments (diamonds) and correspondence relationships (solid arrows) between matched alignments

highlighted initial alignment (blue) was not represented at the target scale Fig. 6.11(a)), the system is able to inform users that the constraint is violated. Another alignment in Figure 6b is kept (see the target alignment highlighted in pink).

Due to the partial matching, however, the assessment of the constraint for an alignment comes with a confidence indicator ($CI_{Existence}$):

$$CI_{Existence}(A_i) = MRatio(A_i) \tag{6.6}$$

where A_i represents the initial alignment to be evaluated. Since the quality statement is directly about initial alignments, the confidence indicator is calculated based on the matching ratio of initial alignments. A lower confidence value means that the system is not sure about the assessment. This may be caused by the fact that the initial alignment is only partly kept, as shown in Fig. 6.11(b), but it may also be the case that some buildings initially composing the alignment were largely displaced, so that they cannot be recognized as part of the target alignment.

To aggregate individual CVs into a global constraint violation for a feature class (building class), weighted average CV (*WACV*) was used. Suppose all initial alignments are equally weighted, the global constraint violation is solely determined by the removed alignments. The more alignments are removed during generalization, the more the constraint is violated for the building class ($CV \rightarrow 1$). Of course, the evaluation should also consider the visual significance of the alignments. For example, removing a less signifi-

(a) Constraint violation (b) Confidence indicator

Figure 6.11 Popup window informing users of detailed information (only problematic alignments are highlighted)

cant alignment is better than removing an important alignment. Hence we define $WACV$ as:

$$WACV = \frac{(Significance\,(A_i) \times CV\,(A_i))}{\sum_{i=1}^{n} Significance(A_i)} \tag{6.7}$$

where $WACV \in [0, 1]$; $Significance(\cdot)$ is defined in Equ. 5.6. A higher value of Equ. 6.7 indicates a higher degree of constraint violation. $WACV = 1$ indicates that the constraint is completely violated for the whole class. For the constraint discussed here, i.e., 'keep the existence of alignments', a complete violation is caused only because all the alignments in the initial building class were removed during generalization. Note also that Equ. 6.7 is used to aggregate the individual assessments of the subsequent constraints.

Alignment orientation

The constraint 'Target orientation should be similar to initial orientation' imposes a condition at the property level, so the evaluation process only inspects the orientations of the matched target alignments with respect to their corresponding initial alignments (detailed explanation refers to Section 3.2.3). However, to compare the orientation of an entire target alignment with respect to the corresponding initial alignments is meaningless, because the target alignment is sometimes very much different from the initial ones, where partial matching and n-to-1 relationship is possible

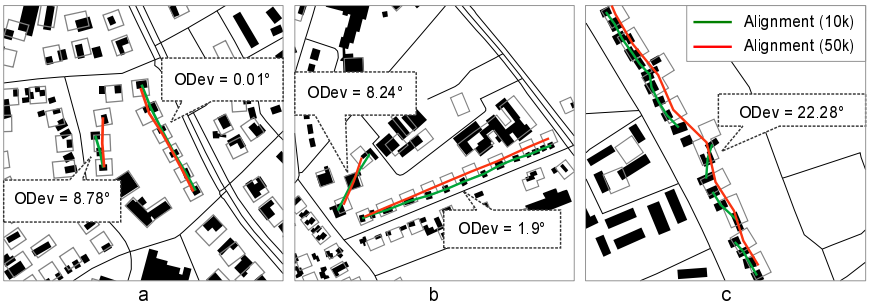


Figure 6.12 Examples of corresponding alignments and measured orientation deviations (a)–(c) (only related alignments are visualized)

(Fig. 6.12(c)). To address this issue, we compare the orientation of the initial alignment with its corresponding parts on the target alignment. This way the n-to-1 comparison is decomposed into several 1-to-1 comparisons and hence initial alignments become the subject of the evaluation of this constraint. Orientations are measured from initial alignments and their corresponding parts on target alignments using method described in Section 5.2.2. The difference in alignment orientation used in this evaluation is defined as follows.

$$ODev(A_i, A_t) = |AlignO(A_i) - AlignO(A_{tp})| \tag{6.8}$$

where A_i and A_t are corresponding alignments; A_{tp} indicates the parts of A_t to which A_i corresponds; $ODev \in [0, \pi/2]$. Finally the deviation is converted into a continuous range [0, 1] in Equ. 6.9, indicating the degree of violation. Since human eyes can hardly tell a small deviation (e.g. 1°) from zero, a tolerance T_1 is introduced to allow for a degree of flexibility. T_2 is introduced to define acceptable deviations. If the deviation is larger than T_2, the constraint is considered to be completely violated; values smaller than T_2 can be converted to different degrees of violation.

$$CV_{Orientation}(A_i) = \begin{cases} 0, & \text{if } ODev \leq T_1 \\ 1, & \text{if } ODev \geq T_2 \\ \frac{ODev - T_1}{T_2 - T_1}, & \text{otherwise} \end{cases} \tag{6.9}$$

As explained above, in evaluating the preservation of alignment orientation, we use initial alignments as the subject of the constraint to simplify the problem caused by partial matching and n-to-1 relationship. Therefore, the calculation of confidence indicator for this constraint is the same as the one for the previous constraint.

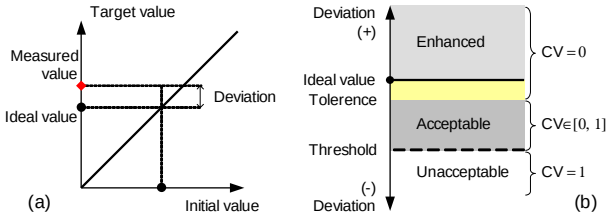


Figure 6.13 Preservation evolution function (a) and interpretation function (b)

$$\mathrm{CI}_{Orientation} = MRatio(A_i) \tag{6.10}$$

where A_i represents the initial alignment to be compared in terms of orientation. Additionally, individual assessments are aggregated for the building class using Equ. 6.7, with the matched initial alignments as the population. Fig. 6.12 shows examples of matched alignment pairs and their orientation deviations. In Fig. 6.12(c), a relatively large *ODev* was identified between the initial alignment and its corresponding part on the target alignment. The large discrepancy in orientation indicates that the orientation of this initial alignment is not well kept. Nevertheless, because the initial alignment is of low significance (too short compared to the target alignment), this violation will not impact too much the aggregated violation.

Spatial distribution of composing buildings

This constraint requires the spatial distribution of buildings in any target alignment be similar to buildings in the corresponding initial alignment(s). To deal with n-to-1 relationship occurring in evaluating this constraint, we do not segment target alignments into corresponding parts, since some parts may relate to less than three objects from which homogeneity cannot be measured (Section 4.1.4). Instead, we firstly summarize the homogeneities of the n initial alignments into a single value and then compare this value with the target homogeneity. To achieve this, initial alignments involved in a n-to-1 relationship are weighted by the number of composing objects and an average homogeneity can be computed.

The evaluation of the similarity of the distribution can be formalized using the preservation evolution function [21]. The preservation function (Fig. 6.13(a)) describes that the ideal state (homogeneity) of a target alignment is equal to its initial state. Actually, a deviation between the target and ideal value can always be observed in reality, which is defined as:

$$HDev(A_t, CS(A_t)) = Homo(A_t) - Homo(CS(A_t)) \tag{6.11}$$

The corresponding relationship between A_i and A_t is specified in function $C(A_i, A_t)$; $C(A_i, A_t) = 1$ if A_i corresponds to A_t and $C(A_i, A_t) = 0$ if otherwise. Then we can define $CS(A_t) = \{A_i | C(A_i, A_t) = 1\}$. $Homo(CS(A_t))$ is the above-mentioned averaged homogeneity of corresponding alignments of A_t. $HDev > 0$ indicates homogeneity increases after the generalization. That is when A_t becomes more regular (enhanced) and the constraint is regarded to be fulfilled ($CV = 0$). $HDev < 0$ indicates homogeneity decreases (when A_t becomes less regular). When such a decreasing is smaller than a tolerance T_3 (a small amount of change that can be ignored by human eyes), the regularity is still considered to be similar ($CV = 0$). When the decreasing in homogeneity exceeds the threshold T_4 (which defines an acceptable amount of change), the constraint is completely violated ($CV = 1$). The values in between gives different degrees of violation. Note that T_3 and T_4 are both negative and $T_3 > T_4$. These conditions are described in Fig. 6.13(b) and can be formally defined as follows:

$$CV_{Dist}(A_t) = \begin{cases} 0, & \text{if } HDev \geq T_3 \\ 1, & \text{if } HDev \leq T_4 \\ \frac{HDev - T_3}{T_4 - T_3}, & \text{otherwise} \end{cases} \tag{6.12}$$

Again, users should be informed of the confidence of the evaluation decision. The homogeneities are measured based on both initial and target alignment rather on the corresponding parts. As a result, *MRatio* of both initial and target alignment in a correspondence relationship contribute to the measure of confidence for this constraint (CI_{Dist}):

$$CI_{Dist} = MRatio(A_t) \times \frac{\sum_{i=1}^{n} MVNum(A_i)}{\sum_{i=1}^{n} VNum(A_i)} \tag{6.13}$$

where A_i and A_t follows $CS(A_t) = \{A_i | C(A_i, A_t) = 1\}$. The summation symbol of the second term is a general formation to allow for n-to-1 relationship: when $n = 1$, this second term becomes $MRatio(A_i)$; otherwise the second term in Equ. 6.13 is determined by the n initial alignments. Equ. 6.13 has the following property: when both initial and target alignments are partially matched to each other (e.g. those in Fig. 6.6(b)), the equation doubles the uncertainty of the evaluation result.

The alignment pair (Fig. 6.14(a), left) shows a slight decrease in homogeneity ($HDev < 0$), indicating that the target alignment is less regular than the initial one. But this pair seems to be a good generalization to human eyes. In the interpretation phase, this is handled by tolerance T_3 so that $CV = 0$. The other pair (Fig. 6.14(a), right) shows an enhanced alignment with increased homogeneity. Fig. 6.14(b) exemplifies a case where users can be informed of

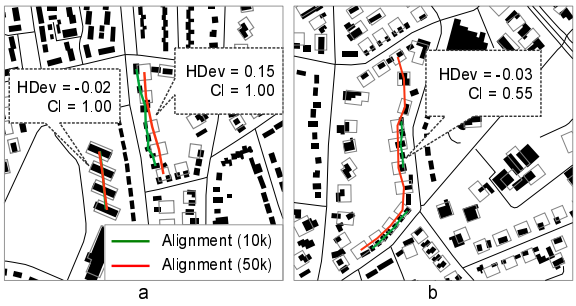


Figure 6.14 Examples of corresponding alignments, homogeneity deviations, and confidence indicator

a low confidence (0.55) due to partial matching. This shows further that the undesired decrease in homogeneity may not be caused by a poor generalization but may be the result of partial matching and n-to-1 correspondence.

Again Equ. 6.7 is used to aggregate individual assessments of this constraint. To be useful, A_i in Equ. 6.7 should be replaced by A_t since target alignments are now the subject of this particular constraint, where the population is the matched target alignments.

With this presented evaluation approach, we will evaluate real topographic datasets against the above three constraints on alignments in Section 7.5.

6.5 Evaluation of spatial distribution

In Section 5.6, we have described a distance-based approach to measuring the similarity between spatial distributions of locations. Although the approach can be applied to point and areal features, it cannot be used to measure the similarity between distributions of linear features such as networks. On the other hand, this approach is not capable of describing the density balance as required by conventional specifications for map generalization (see e.g. SSC [184]). In this section, we present an approach to evaluate the spatial distribution as well as density balance of map objects based on local densities (Section 5.5.1).

In Section 5.5.1, we have argued that the spatial distribution of a group of objects (a meso-level group or a feature class) can be reflected by the distribution of local densities. After establishing the corresponding relationships between individual objects (Section 6.2), their spatial distribution can be evaluated by examining the balance of the distribution of their densities. To evaluate whether or not density balance is preserved, we compare initial densities with their corresponding target densities. The objective

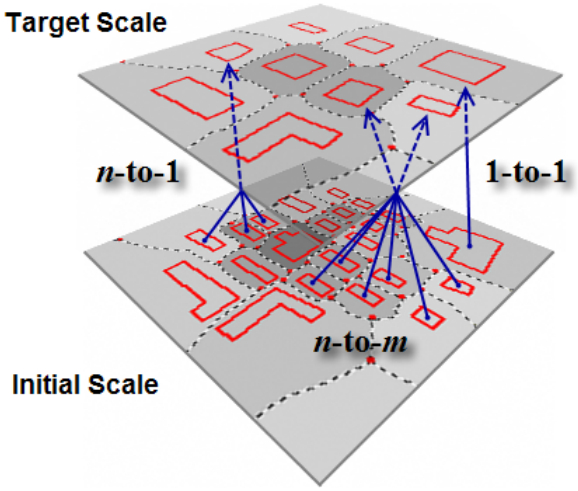


Figure 6.15 Corresponding relationships between initial and target objects with local densities

of the constraint on keeping spatial distribution (or density balance) is that dense areas should remain dense, and differences in densities should also be retained. In more formal terms, target densities should correlate to initial densities (or the variance of the change of local density should be minimized). In the following, we firstly describe how to deal with many-to-many corresponding relationships (Section 6.5.1) and then present how to evaluate the similarity of spatial distribution based on local densities (Section 6.5.2).

6.5.1 *Handling n-to-1 and n-to-m relationships*

Formally, we use Equ. 6.14 to denote the comparison of corresponding local densities:

$$Comp(Density(E_i^{S_I}), Density(E_i^{S_T})) \tag{6.14}$$

where $E_i^{S_I}$ and $E_i^{S_T}$ represent initial and target entities that correspond to each other. According to formal theory presented in Section 4.1.1, entities could be an individual object, a group of objects, or a feature class. Consequently, different multiplicities are possible between corresponding individuals (see Fig. 6.15). It is no problem to compare the change of local densities for 1-to-1 correspondence, but it would be problematic for n-to-1 and n-to-m correspondences.

To address this issue, we aggregate the densities of the n (or m) objects forming the meso-entity into a so-called meso-density:

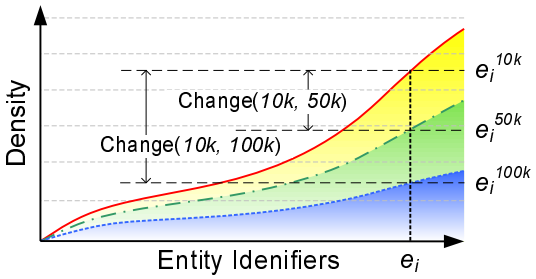


Figure 6.16 Relating the change of densities to scale reduction (the values related by the vertical dashed line are local densities of corresponding entities at different scales)

$$\textit{Meso-Density}\left(E_i^S\right) = \frac{\sum_{i=1}^{n} \textit{Density}(o_i) \times \textit{ImpactA}(o_i)}{\sum_{i=1}^{n} \textit{ImpactA}(o_i)} \tag{6.15}$$

where o_i represent every object composing the meso-entity E_i^S; *Density*$(\cdot)$ and *ImpactA*$(\cdot)$ are local density and impact area discussed in Section 5.5.1. This equation is in essence a weighted average where the impact areas of involved objects are used as the weighting factor. It can be applied to both the initial and target scales, so that after the aggregation Equ. 6.14 can be viewed as a 1-to-1 comparison between spatial entities. Take n-to-m relationship for example, Equ. 6.15 can be applied to both the n initial objects and the m target objects, after which the comparison becomes the comparison of two meso-densities.

6.5.2 *Similarity evaluation by correlation analysis*

As mentioned previously, the similarity of spatial distributions can be evaluated with correlation analysis of corresponding densities. To prepare data for the analysis, we start from target densities and their corresponding initial densities. Then we aggregate local densities into meso-densities in the case of n-to-m relationship. Finally the densities is converted to a two-column data for correlation analysis. The higher (approaches to 1) the resulting correlation value, the more similar the two distributions of densities are. A value of zero indicates that there is no correlation between the two datasets, meaning that they are very dissimilar and the initial spatial distribution is not well preserved in the target dataset. A value in between can be used to indicate the degree of constraint satisfaction. In addition to the analysis, plot and scatter plot can be intuitively insightful.

However, correlation analysis only indicates that the two distributions are similar. It is usually the case that spatial distributions can be similar but contain different number of objects. This means that correlation analysis is

a relative measure and cannot relate the change to scale reduction. Ideally, the change of local densities should be related to the reduction of scales, so that for each scale transition a reference change ratio of density can be determined (see e.g. Fig. 6.16). In practice, as will be shown in Section 7.6.1, target densities may not strictly follow a fixed change ratio. So variance of real change ratios can be used to assist the analysis. Also with this method, exceptions in the change of densities can also be identified and spotted on the map for cartographers to check why and how it is the case.

To summarize, correlation analysis gives indications to the relative similarity between spatial distributions, while change ratio relates map specifications to scales. Both are equally valuable. In Section 7.6.1, we implemented the approach described in this section and present some results.

7 Implementation and validation

7.1 INTRODUCTION

This chapter implements the key concepts in the conceptual framework (Chapter 3) and the system components of the design (Chapters 5 and 6), and reports results of the implementation and experiments. Chapter 4 gives a formal framework for all the key aspects and components including a precisely definition of some of the constraints to be automatically evaluated in this chapter. Instead of realizing a functioning evaluation system, we implemented the key aspects and components separately.

7.1.1 *Implementation summary*

The major components of the proposed evaluation framework were implemented as extensions to DoMap[1] – an interactive generalization platform that has been implemented and maintained in `Visual C++ 6.0` by our colleagues at Wuhan University since 1998. The platform was extended and used in a national wide project in China to generalize and update topographic databases (from 1:10k to 1:50k).

The data model of DoMap has evolved from geometry/topology-first to object-first approach (see Stoter et al. [195] for a discussion of the modeling approaches in the geospatial domain). The object-first approach models object classes (including geometries) first which are then added with thematic attributes. In DoMap, thematic attributes are stored in a relational database per dataset and linked to objects by key values. Every feature class (layer) has independent descriptions of geometries and attribute definitions. In current model of DoMap, topology is not explicitly stored but is calculated on demand. This is to accelerate the interactive visualization and data editing. The same strategy is also applied in ESRI's Geodatabases.

[1] DoMap has also a commercial version named GenTools

In the functional aspect, DoMap provides basic geometric and topological operations as well as an implementation of Delaunay triangulation, based on which we implemented the many concepts described in Chapter 5. The approaches described in Chapters 5 and 6 were also implemented on top of DoMap.

In the following, we present the implementations of building clustering (Section 7.2), building alignment recognition (Section 7.3), data matching of buildings (Section 7.4), matching of building alignments (Section 7.5.3), evaluation of preservation of alignments (Section 7.5) and preservation of spatial distribution (Section 7.6).

OWN PUBLICATIONS

Results reported in this chapter are partly based on the following publications:

- X. Zhang, T. Ai, and J. Stoter. The evaluation of spatial distribution density in map generalization. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Congress 2008)*, volume XXXVII. Part B2., pages 181–188, Beijing, 2008.
- X. Zhang, X. Zhao, M. Molenaar, J. Stoter, M.-J. Kraak, T. Ai. Pattern classification approaches to matching building polygons at multiple scales. In *(ISPRS Congress 2012)*, Melbourne, 2012 (to appear).
- X. Zhang, T. Ai, J. Stoter, M.-J. Kraak, and M. Molenaar. Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. *GeoInformatica*, page 33 (online first). doi: 10.1007/s10707-011-0146-3.
- X. Zhang, J. Stoter, T. Ai, M.-J. Kraak, and M. Molenaar. Automated evaluation of building alignments in generalised data. *Manuscript submitted for review to an academic journal*.

7.2 BUILDING CLUSTERING

In this part, we implement the building clustering method described in Section 5.3.3, where the concepts of refined CDT (Section A.1), MST, and inconsistent edge (Section 5.3.3) are realized. An experiment was carried out to tune the parameters involved in the definition of inconsistent edges, and some results are shown in Fig. 7.1 and 7.2. The clustering method was

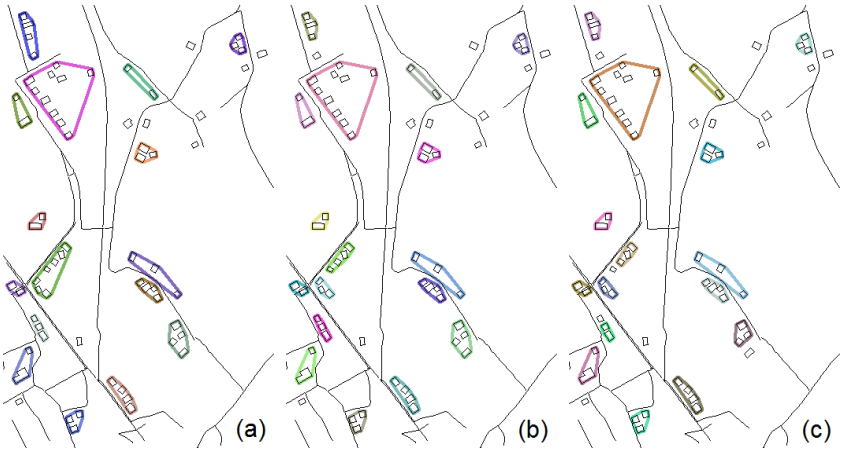


Figure 7.1 Building clustering with $n = 3$ and $f = 3$ (a), 2.4 (b), and 2 (c) (Data source: Kadaster, NL)

tested against datasets of the Netherlands and China.

The involved parameters are mainly n and f in Equ. 5.5 that define what is meant by that an edge is *'significantly larger than'* its neighboring edges. The parameter f is in effective when the standard deviation of neighboring edges approaches to zero. In Fig. 7.1 we fix n and adjust f, the clustering results show only some minor differences. When we change n, the parameter that control variance, we observe that with the decrease of n we get more homogeneous clusters (see e.g. how the biggest cluster in Fig. 7.2(a) changes). This is because with the decrease of n, the criterion to select inconsistent edges is more relaxed. For example, the length criterion for an edge (e_i) to be significantly larger than its neighboring edges is $\text{Length}(e_i) > \text{mean}_{\text{weight}} + 3 \times \text{std}_{\text{weight}}$ given $n = 3$, while the criterion is $\text{Length}(e_i) > \text{mean}_{\text{weight}}$ given $n = 0$. This means that when we decrease n, more edges would be identified as inconsistent edges, and the remaining clusters are more homogeneous.

Another important observation is that, the performance of the clustering method depends on the data characteristics. For the buildings that are in general clustered, as in Fig. 7.2(d)–7.2(f), the clustering method fails to detect meaningful building patterns no matter how to tune the parameters.

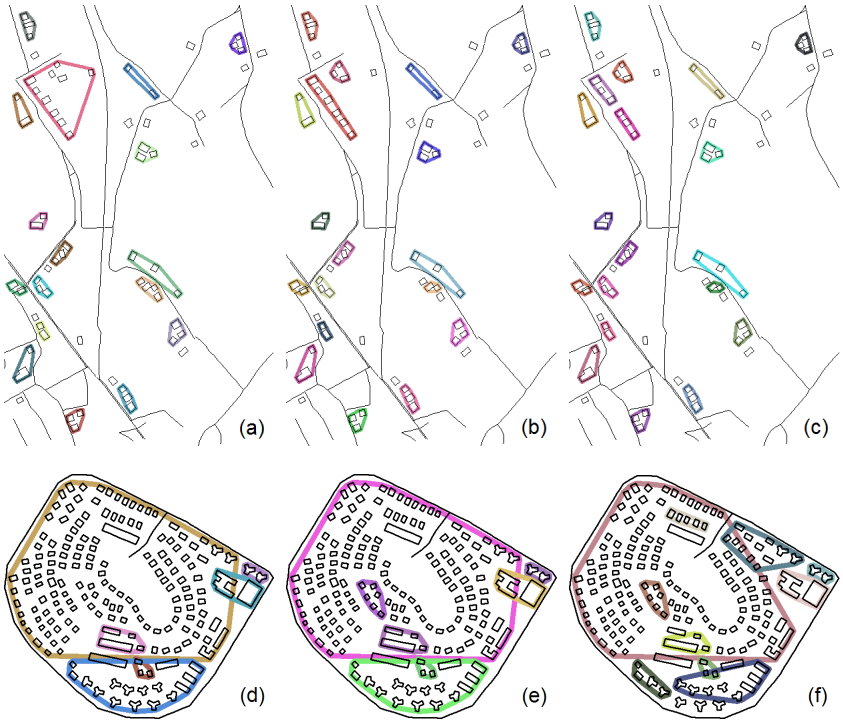


Figure 7.2 Building clustering with $f = 2$ and $n = 3$ (left), 2 (middle), and 1 (right) (Data source: Top – Kadaster, NL , Bottom – Shenzhen, China)

7.3 Recognition of building alignments

To recognize building alignments, we implemented four recognition algorithms, including two algorithms for collinear alignments using alignment-of-center (*CollAlgo1*) and alignment-of-side (*CollAlgo2*), as well as two algorithms for curvilinear alignments using alignment-of-center (*CurvAlgo1*) and alignment-of-side (*CurvAlgo2*). Note that the algorithms are based on the clustering method with $f = 2$ and $n = 3$. We tested the recognition framework described in Section 5.4 against several topographic datasets of different geographic areas and scales. Dataset1 (scale 1:10k), Dataset2, and Dataset3 (scale 1:50k) are part of Dutch topographic datasets (Kadaster, the Netherlands); Dataset4 is part of ICC (Catalonia, Spain) topographic datasets at 1:25k; Dataset5 is part of the 1:10k topographic data of China. Fig. 7.3 demonstrates a general impression of some algorithms. Fig. 7.3(e) and 7.3(f) also shows that the alignment recognition algorithms explicitly

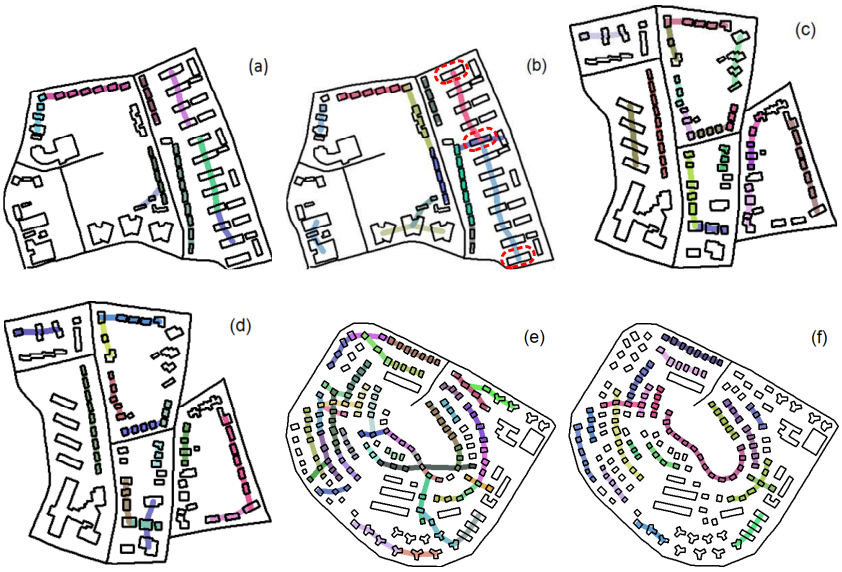


Figure 7.3 Results obtained by *CollAlgo1* (a), *CollAlgo2* (b), *CollAlgo1* (c), and *CurvAlgo1* (d) on Dataset1 (Data source: Kadaster, NL), and results by *CollAlgo1* (e) and *CurvAlgo1* (b) on Dataset5 (Data source: Shenzhen, China)

identify the alignments as oppose to the clustering method (cf. Fig. 7.2(d)–7.2(f)).

Four experiments were carried out to show the result of parametrization (Section 7.3.1), application of the recognition framework (Section 7.3.2), recognition quality (Section 7.3.3), and curvature descriptor (Section 7.3.4). Note that in the resulting figures black thin lines (if applicable) denote MST edges and (randomly) colored bold lines denote recognized patterns. After that the results are discussed in Sections 7.3.5–7.3.7.

7.3.1 *Adaptation of parameters*

To demonstrate the algorithms, different parameter values were tested to show their influence on the recognition results. We found that the most dominant parameters are *PathAngleDev* for collinear pattern and *MaxPathAngle* for curvilinear pattern. First, *PathAngleDev* = {5°, 10°, 15°, 20°, 25°, 30°} were used to detect collinear alignments in Dataset2 (Fig. 7.4).

Fig. 7.4 shows that the best working range of *PathAngleDev* for collinear

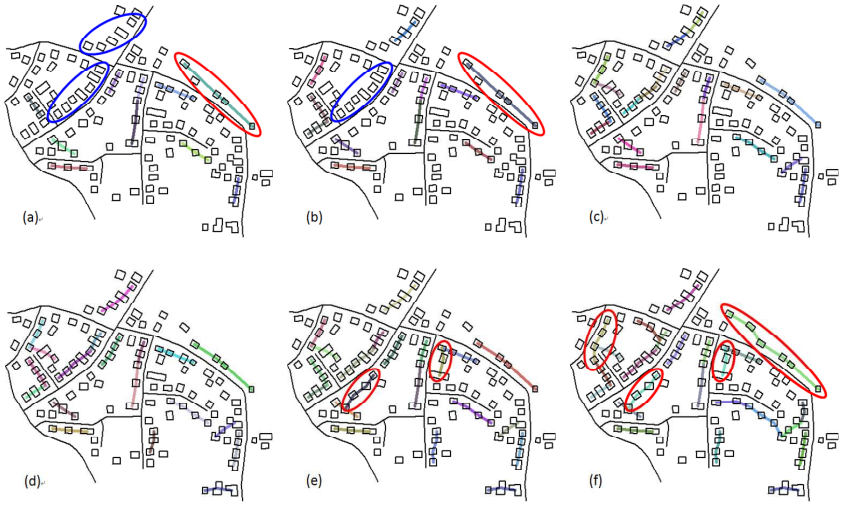


Figure 7.4 Result of *CollAlgo1* with different path angle values: (a) 5°, (b) 10°, (c) 15°, (d) 20°, (e) 25°, (f) 30°; obvious false positives (red circles) and false negatives (blue circles) are exemplified (Dataset2: Kadaster, NL)

algorithms was from 15° to 20°. When using a value lower than 10°, some alignments were missed (false negatives, blue circles in Fig. 7.4). When using a threshold higher than 25°, some false positives were created (red circles in Fig. 7.4). The tests on the other datasets confirmed this working range of *PathAngleDev*.

Likewise, the *MaxPathAngle* = {40°, 60°} was used for curvilinear patterns. The two values for our test cases did not show any difference, except for the alignment highlighted by the red circle on the bottom of Fig. 7.6(b) which can only be detected with the value of 60°. Consequently *MaxPathAngle* = 60° is preferred since it is more tolerable to alignments with bigger curvatures.

7.3.2 *Application of the recognition framework*

To demonstrate the recognition framework (Section 5.4), including the combination step (Section 5.4.6), the following parameters were used for all the datasets: *PathAngleDev* = 18°, *OrientationDev* = 45° (for collinear), *MaxPathAngle* = 60° (for curvilinear), and *SizeContrast* = 3.2 (used for alignment-of-center technique only). Note that the four component homogeneities (Equ. 5.2) were equally weighted to calculate the integrated

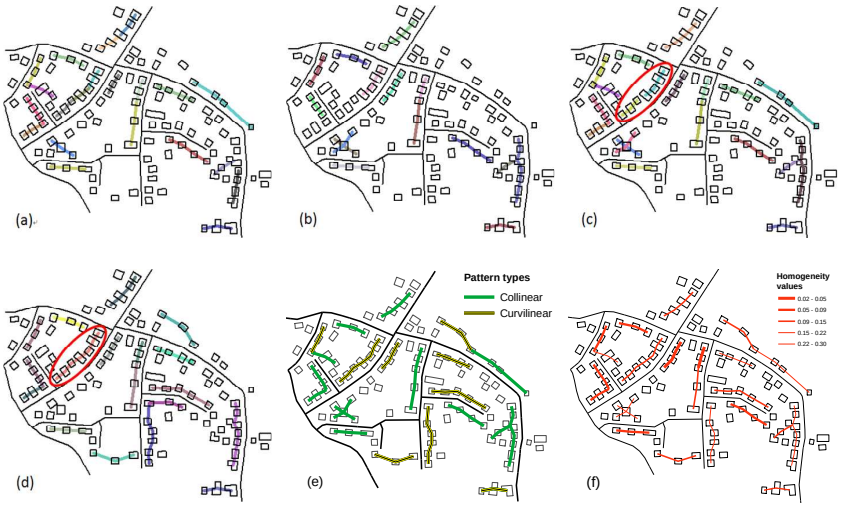


Figure 7.5 Combination of different results: (a) *CollAlgo1*; (b) *CollAlgo2*; (c) harmonized result for (a) and (b); (d) harmonized result for *CurvAlgo1* and *CurvAlgo2*; the final combined result visualized by pattern types (e) and homogeneity values (f); (Dataset2: Kadaster, NL)

homogeneity (Equ. 5.3). Fig. 7.5, Fig. 7.6 and Fig. 7.7 show intermediate and final combined results for the test datasets.

The first observation in these figures is that, in most situations, all individual algorithms were able to detect alignments that are visually significant. Still, some alignments were not fully recognized by one and other algorithms. It was obvious that collinear algorithms segmented long curves (with apparent variations in curvature) into small pieces of straight lines. Not surprisingly, such long curves were successfully recognized by curvilinear algorithms. For example, the long curves (highlighted by red circles in Fig. 7.5(d) and Fig. 7.6(b)) were detected by curvilinear algorithms, but they were segmented by collinear algorithms (see their counterparts in Fig. 7.5(c) and Fig. 7.6(a)). An extreme case is demonstrated in Fig. 7.7(b)–7.7(d), where most curves were segmented by collinear algorithms. On the other hand, although curvilinear algorithms were also able to detect part of the collinear alignments, they did not recognize all of them. For instance, collinear algorithms recognized the alignments (highlighted by red circles in Fig. 7.6(a) and Fig. 7.7(d)) while curvilinear algorithms did not, as can be seen from their counterparts in Fig. 7.6(b) and Fig. 7.7(b) and 7.7(c).

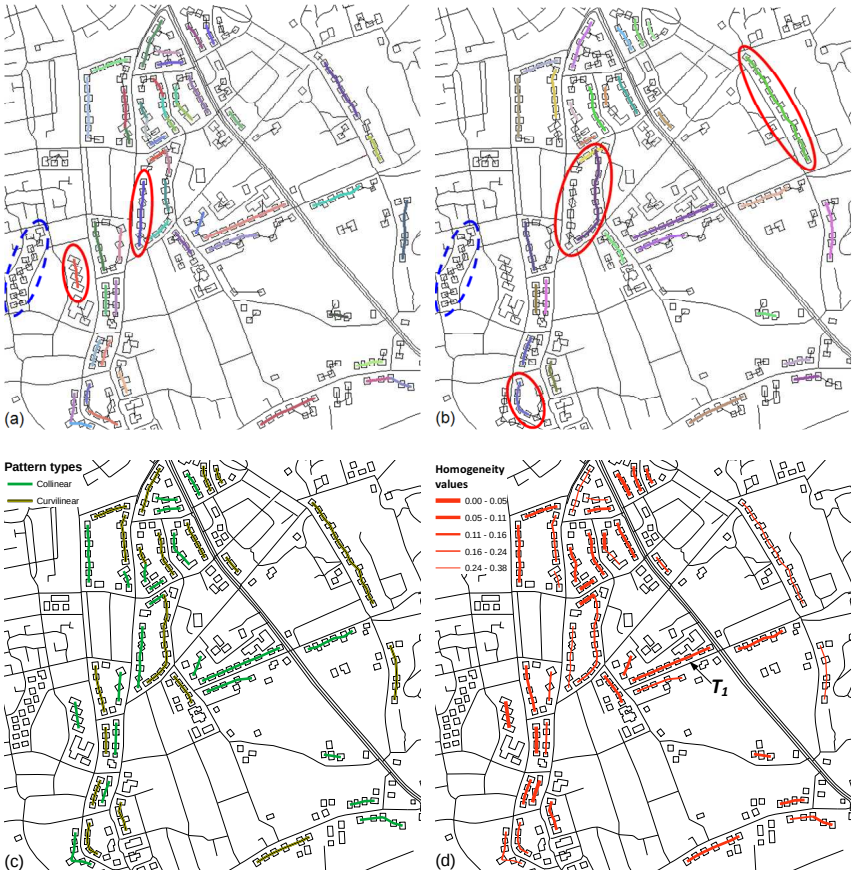


Figure 7.6 Combination of different results: (a) harmonized result for *CollAlgo1* and *CollAlgo2*; (b) harmonized result for *CurvAlgo1* and *CurvAlgo2*; the final combined result visualized by pattern types (c) and homogeneity values (d); (Dataset3: Kadaster, NL)

The difference between algorithms working with alignment-of-center and alignment-of-side techniques is obvious. Examples refer to the differences between e.g. Fig. 7.3(a) and Fig. 7.3(b), and between Fig. 7.5(a) and Fig. 7.5(b). In particular, the areas highlighted in Fig. 7.7(b) identify potential longer curves (curvilinear alignments). These alignments were not fully recognized using alignment-of-center, but they were recognized using alignment-of-side technique (cf. Fig. 7.7(c)).

Last but not least, the experiment shows that the final results combined

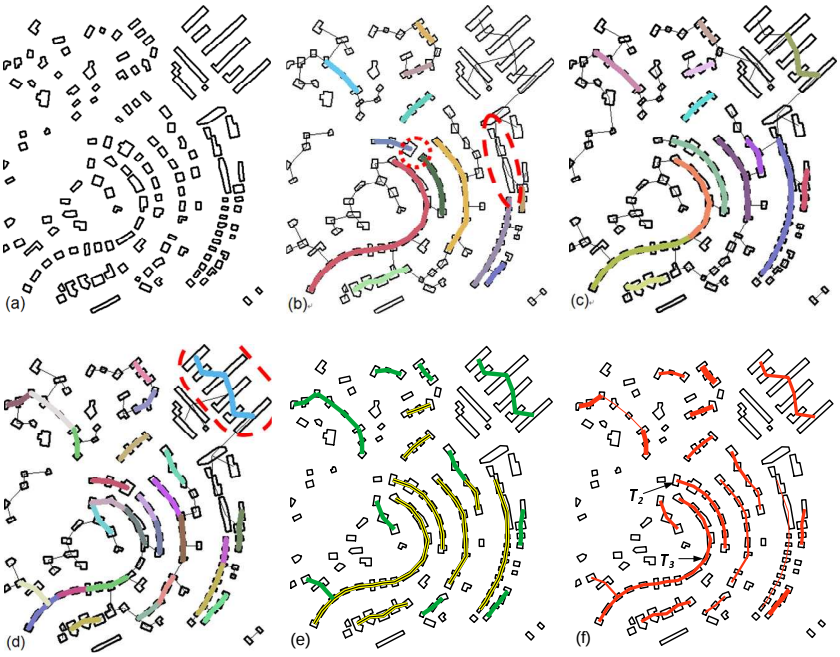


Figure 7.7 Combination of different results: (a) Dataset4; (b) *CurvAlgo1*; (c) *CurvAlgo2*; (d) harmonized result for *CollAlgo1* and *CollAlgo2*; the final combined result visualized by pattern types (e) and homogeneity values (f) (Data source: ICC, Catalonia, Spain)

from different algorithms were more complete than the individual results. In fact, they are unions of the individual results. Furthermore, Fig. 7.5(e), Fig. 7.6(c) and Fig. 7.7(e) show that the datasets are mixtures of different types of alignments. Fig. 7.5(f), Fig. 7.6(d) and Fig. 7.7(f) show the homogeneities calculated for the final patterns (the wider the symbol the stronger the pattern is in terms of homogeneity).

7.3.3 *Validation of the recognition results*

To validate the quality of the recognition results, an expert evaluation was carried out based on the survey presented in Appendix B. In this survey, an independent cartographer (from Kadaster, the Netherlands) visually identified the patterns in representative excerpts of the datasets used in this research (Fig. 7.8).

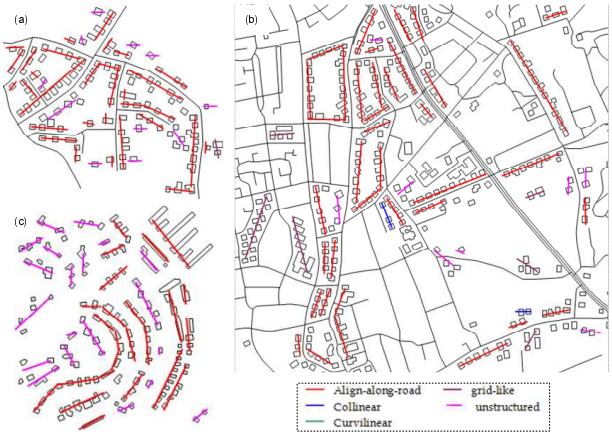


Figure 7.8 Building patterns visually identified by a cartographer (Dataset 2–4)

The survey was presented in a Word document. The invited expert used the drawing tools available in MS Word to 'draw' the patterns he regards to be significant. Note that the expert only used a straight line tool to mark all pattern types, with colors indicating different types of patterns.

We validate the recognition results by comparing the automatically detected alignments (ADAs) with the visually identified alignments (VIAs) and considering the following three side notes. First, for all the three datasets, the expert identified isolated and pairs of buildings as meaningful patterns, but by definition the building patterns in our approach contain at least three buildings. Consequently, we ignored all the VIAs that only contain one or two buildings. Second, since this research focuses on linear alignments, we also ignored grid-like and unstructured patterns. Finally, in the expert's view, curvilinear and align-along-road alignments are variations of collinear alignments. As a result, linear alignments are directly compared regardless of their respective types.

The comparison approach works as follows: if an ADA partly coincide with a VIA (i.e. the two have common features), we count it as a true positive (tp); if an ADA occurs without a (partly) coincident VIA, we count it as a false positive (fp); if a VIA occurs without a (partly) coincident ADA, we count it as a false negative (fn). Then we use precision $= \mathrm{tp}/(\mathrm{tp} + \mathrm{fp})$ and recall $= \mathrm{tp}/(\mathrm{tp} + \mathrm{fn})$ to characterize the recognition quality (Table 7.1).

From the comparison result we can conclude that the recognition quality is in general satisfactory. The recall for the three datasets is high (approx. 90% and higher). This indicates that the automatic approach recognized almost

	Precision	Recall
Dataset2	73.9%	85.9%
Dataset3	77.3%	91.9%
Dataset4	60.0%	100.0%

Table 7.1 Precision and recall of automatically recognized alignments with respect to visually identified alignments (Dataset2-4)

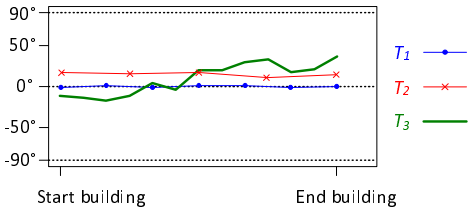


Figure 7.9 Distribution of local curvatures measured for sample alignments

all the VIAs. The relatively low precision (lower than 80%) indicates that the automatic approach recognized more alignments than the expert. That the expert chose more rigorous rules in determining the alignments could be one possible reason. The ambiguities in defining and distinguishing the patterns could be another, which we will discuss in Section 7.3.6.

7.3.4 *Curvature descriptor demonstrated*

As described in Section 5.4.5, the curvature descriptor can be used to enrich the description of building alignments. This enrichment is useful in automated generalization of building alignments for keeping their specific shapes. For illustration, the descriptor was applied to some detected linear alignments (*T1*, *T2* and *T3* marked in Fig. 7.6(d) and Fig. 7.7(f)). According to the curvature descriptor, we can identify alignments with straight-line shape (*T1*), constant curvatures (*T2*) and wavy curvatures (*T3*) in Fig. 7.9, which agrees with their respective spatial arrangements.

7.3.5 *Single algorithm versus multi-algorithm approach*

The results show that the collinear and curvilinear algorithms and their variations (e.g. alignment-of-side technique) are able to recognize building patterns that are also contented with human perception. However, these individual algorithms are not competent to fully recognize building alignments. The results show merits and limitations of each algorithm and that different algorithms are partly complementary. Therefore a framework that combines different algorithms will yield better results for building patterns

recognition than the single algorithm approach, which can be concluded from several observations.

Potentials and problems of the individual algorithms

The algorithms for collinear and curvilinear patterns are appropriate to recognize straight lines respectively curves. These individual algorithms refine the general clusters (clusters linked by black thin lines in Fig. 7.6(a)) detected by an approach similar to [156]. In the design phase, we made a sharp distinction between collinear and curvilinear patterns. It is thus reasonable that the collinear algorithms fail to recognize curvilinear alignments. On the other hand, straight lines can be regarded as a special case of curves, so the curvilinear algorithms are expected to be able to find collinear alignments. The experiments confirm that, this is true in many cases but not in all cases. For example, when the buildings align themselves to the same direction, and when this direction is quite different from the normal direction of the path (e.g. the red circles in Fig. 7.6(a)), curvilinear algorithms will identify them as misaligned buildings. However such alignments can be successfully detected by collinear algorithms, because they use a much looser rule on building orientations. Given this deficiency, this study shows that still the visual perception theory [67] used in the curvilinear algorithms is effective and yields satisfactory results, especially in recognizing long curvilinear patterns.

On the other hand, the results demonstrate that it is very difficult to use a single algorithm to detect all patterns. First of all, universal rules appropriate for detecting different pattern types are not available. Usually different ad-hoc models are used. Technically, it is also difficult to accommodate competing recognition rules (e.g. for collinear, curvilinear, alignment-of-center and alignment-of-side) in one algorithm. Considering the fact that the recognition results from different algorithms are partly complementary, using multiple algorithms followed by a combination process should yield better results. The experiments that combined the results of individual algorithms confirmed this (see Fig. 7.5, Fig. 7.6 and Fig. 7.7).

Advantages of using a multi-algorithm paradigm

The use of multiple algorithms followed by a combination process also improves the quality (in terms of homogeneity) of the detected patterns. This is because, in the combination process, two conflicting patterns are evaluated against each other, and the one with better quality (defined in terms of number of elements and homogeneity, see Section 5.4.6) is selected

as the final result.

Admittedly, each specific algorithm may be well suited for a certain type of data. But real data are mostly mixtures of various characteristics as shown in the experiments, i.e., only in rare cases a dominant pattern type may character a dataset. Even if a dataset is rather homogeneous, with the single algorithm approach one still has to identify the main characteristics of the dataset and manually choose the most appropriate algorithm. The proposed multi-algorithm approach can deal better with these problems due to the evaluation and combination mechanism. Therefore, it is more adaptive to different types of data.

7.3.6 *Ambiguities involved*

Several observations from the expert survey (Section 7.3.3) demonstrate further ambiguities outlined in Section 5.3.1. An example of the ambiguity in defining building patterns is how many buildings counts as a pattern, as described in Section 7.3.3. Besides, the groups that were identified by both the expert and the proposed approach differ slightly. The difference lies in whether a building belongs to a group or not (e.g. the red circles in Fig. 7.6(a) and their counterparts in Fig. 7.8(b)). The reason may be the ambiguity in the definition, which causes the use of different recognition rules and parameters.

The survey result also demonstrates ambiguities in distinguishing pattern types. Typically, the expert identified linear alignments as align-along-road patterns. The reason, also inferred from the expert survey, was the presence of either real or virtual roads (e.g. white spaces between alignments in Fig. 7.8(c)). The expert may therefore characterize linear alignments as unstructured clusters where he cannot perceive virtual roads. For instance, he identified obvious linear alignments as unstructured clusters in Fig. 7.8(c), which explains why the precision for dataset4 is the lowest (60%). The above observation shows a different understanding of how to distinguish types of patterns. Additionally, the curvilinear alignments in Fig. 7.8(c) were segmented into consecutive straight lines and identified as align-along-road patterns by the expert. This may be due to the limited drawing tools available, but the reason could also be that the boundary between linear types is ambiguous and subjective.

7.3.7 Limitations

One of the drawbacks of our current approach is that it sometimes fails to recognize linear alignments in densely populated areas, such as city centers, commercial and industrial areas. In such areas the MST usually structures as trees with short branches, where linear paths are not clearly visible (see e.g. the group highlighted by the blue dashed circle in Fig. 7.6). Ideally, we would prefer to classify such building groups as nonlinear patterns, because linear characteristics are no longer significant there. As a result, future work is needed to complete our current framework by allowing for nonlinear clusters. A simple solution could be firstly removing the recognized linear alignments from the MST, and then applying e.g. the unstructured pattern detection algorithm described in [238] to recognize nonlinear clusters.

To perfect the recognition process, contextual knowledge should also be incorporated. Actually, different types of building patterns may appear in different places depending on spatial contexts. For instance, linear alignments can be found in urban (e.g. residential areas and partly town centers) and suburban areas; nonlinear clusters are dominant in city centers and industrial areas; random arrangements can be found in scattered and mountain villages [184, 186]. The ratio between the detected alignments and the general clusters may be used to partly imply the context of the built environment, although the determination of the context is a more complex inference, which depends also on size, shape and density of buildings [186].

The ambiguities and subjectivity involved in the expert survey lead to several problems in the evaluation of recognition quality. One problem is the low but problematic precision obtained in Section 7.3.3. Apart from the reason of ambiguities explained in Section 6.3, another reason is the way that the expert dealt with the buildings shared among patterns. In the survey, the expert sometimes precluded a building from an alignment if this building is already part of another alignment (cf. the two horizontal alignments in Fig. 5.14(e) and their counterparts in Fig. 7.8(b)). Because of this, the expert may end up with alignments consisting of only two buildings, which also contributes to the decrease in precision. To address this issue, the survey should be further improved by clearly explaining the concepts and terms used, providing more appropriate ways for experts to identify the patterns. In addition, inviting more experts from different organizations would enable to draw more persistent conclusions.

Note also that some single objects in the removed alignments, usually vertices of degree 1 and 3, may also be part of other patterns (see e.g.

Fig. 7.3(b)). We decided not to keep those shared objects when removing detected patterns for the following reasons. First, the removal of patterns only occurs between iterations, while most alignments with shared objects were detected in the first iteration. Second, false positives would have been created if the shared objects were kept. For example, the three buildings marked in Fig. 7.3(b) would consequently have been recognized as a new alignment which does not really exist. Still, it is true that some potential patterns may be missed if we do not keep those shared objects. To better address this problem without causing new problems, a possible improvement is discussed in Section 8.3.2. Similarly, a group of objects can be part of another group. Nevertheless, this does not contradict our approach. For instance, a collinear alignment detected using *CollAlgo1* may be part of a curvilinear alignment using *CurvAlgo2*. This part-of relation can be used to form building pattern hierarchies. Also, the two alignments can be combined into a harmonized result using the method described in Section 5.4.6.

7.4 Data matching via supervised classification

The generic data matching approach based on pattern classification as described in Sections 6.2 was implemented as follows. First, the similarity measures described in Section 6.2.2 were implemented in DoMap. Measured characteristics of training samples were generated in DoMap and exported to the classifiers (Section 6.2.3). The classifiers were implemented using third party software packages. That is, Naive Bayes and CART based classification was carried out in MATLAB® software[2]; C4.5 was based on the implementation provided by Dr. Ross Quinlan[3] (inventor of C4.5); as for SVM, we used a popular implementation provided in LIBSVM package (see Section 6.2.3) with a Python interface, which also realized an automatic parametrization. Finally, the classification results were transferred back to and visualized in DoMap.

To help human subjects match buildings at different scales, we also developed an interactive matching toolbox in DoMap (Fig. 7.10(a)). This enables to generate training data for the subsequent learning process. In addition, weighted summation was also implemented (Fig. 7.10(b)) to give a general idea of its matching accuracy with respect to the classifiers.

Section 7.4.1 describes how the training samples were generated, after

[2]MATLAB 7.8 (R2009a): http://www.mathworks.com/
[3]C4.5 algorithm: http://www.rulequest.com/Personal/

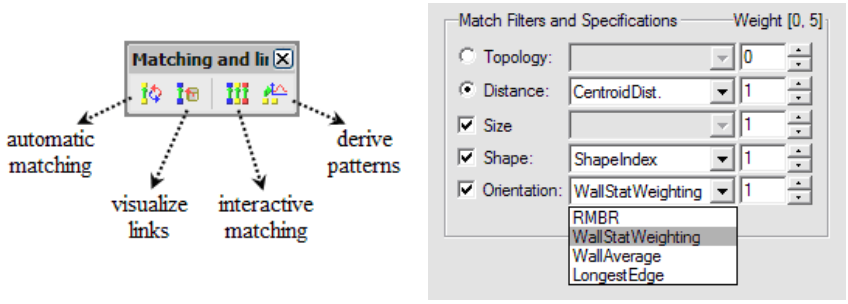


Figure 7.10 Implemented interfaces for interactive matching assist (left) and weighted average approach (right)

	No. of buildings		Training sample
	1:10k	1:50k	No. of pairs (Matched | UnMatched)
Study area A	4272	1720	11578 (1678 | 9900)
Study area B	2646	1637	9445 (1774 | 7671)

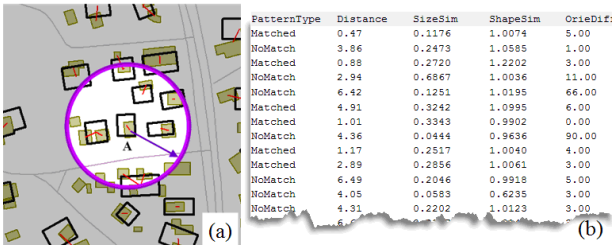
Table 7.2 Overview of topographic data and training samples

which the classifiers were trained and applied to new data (accuracy of the matching is reported and validated in Section 7.4.2). Section 7.4.3 presents problems of current matching approaches and improved results by incorporating domain knowledge into classifiers of soft decisions. Finally, Section 7.4.4 reflects on the matching criteria used.

7.4.1 *Topographic data and training sample generation*

The datasets to be matched are Dutch topographic datasets at 1:10k and 1:50k (i.e. TOP10NL and TOP50NL, Kadaster). In total four datasets are used in this study, i.e., TOP10NL and TOP50NL at study area A and B. This is illustrated in Table 7.2, where the number of buildings in each dataset is shown, along with the generated training samples.

Training data, including positive (matched pairs) and negative (unmatched pairs) classes were generated interactively by human operators. First, the human operators (both have a cartography background) linked buildings in TOP10NL to those in TOP50NL for the two study areas. Only positive samples (see the red links in Fig. 7.11(a)) are generated after this step. Second, starting from each source building (e.g. A in Fig. 7.11(a)) in TOP10NL



PatternType	Distance	SizeSim	ShapeSim	OrieDif
Matched	0.47	0.1176	1.0074	5.00
NoMatch	3.86	0.2473	1.0585	1.00
Matched	0.88	0.2720	1.2202	3.00
NoMatch	2.94	0.6867	1.0036	11.00
NoMatch	6.42	0.1251	1.0195	66.00
Matched	4.91	0.3242	1.0995	6.00
Matched	1.01	0.3343	0.9902	0.00
NoMatch	4.36	0.0444	0.9636	90.00
Matched	1.17	0.2517	1.0040	4.00
Matched	2.89	0.2856	1.0061	3.00
NoMatch	6.49	0.2046	0.9918	5.00
NoMatch	4.05	0.0583	0.6235	3.00
NoMatch	4.31	0.2202	1.0123	3.00

Figure 7.11 Strategy for generating training data (a) and exported training samples (b)

datasets, a batch program searched for matching candidates in TOP50NL that are located within a radius R of this source (the circle in Fig. 7.11(a)); all candidates expect for ones linked interactively by human subjects were recorded as negative samples. The radius R = 6mm (map unit at 1:10k) was used, which is larger than the max distance of all interactively matched pairs.

Finally, two training sets were generated (Table 7.2): one with 11578 pairs of objects between TOP10NL and TOP50NL for study area A (sample set A), and another with 9445 pairs for study area B (sample set B). The training samples were exported into files (e.g. Fig. 7.11(b)) and transferred to the classifiers.

7.4.2 *Validation criteria and matching accuracy*

To validate the pattern classification approaches to data matching, we carried out the following experiments. In general we generated two training sample sets A and B, each of which can be characterized differently. First, to show how well the trained classifiers can be applied to classify novel patterns (new data), we trained the classifiers with sample A and tested with B, and then reverse. This way we also get insight into whether the prediction power of the classifiers relies on spatial characteristics. Second, we trained and tested the classifiers based on each one of the training sets. Third, we trained the classifiers with individual test sets and tested them using 10-folds cross-validation. To get more insight into their performance, different parameter values were tried (Tables 7.3 and 7.4). Therefore, we have different versions of classifiers, including CART*, CART$^{95\%}$, CART$^{85\%}$, SVM$^{0.5,2}$ (detailed explanations refer to Section 6.2.3).

The criteria used to evaluate the performance (accuracy) of different classifiers are *precision* and *recall*, which are widely used indices in classification and information retrieval. Besides, tree size is used as a indicator of tree-based classifiers.

	Setting 1: classifier trained with A and tested with B		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	37.5	78.8	242
CART$^{95\%}$	85.9	75.7	128
CART$^{85\%}$	83.9	80.1	45
C4.5	84.9	81.0	19
NB	84.9	82.0	
SVM$^{0.5,2}$	85.3	79.0	
	Setting 2: classifier trained with B and tested with A		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	75.8	83.3	325
CART$^{95\%}$	82.1	82.4	186
CART$^{85\%}$	80.1	86.3	94
C4.5	85.9	82.9	11
NB	79.0	87.7	
SVM$^{0.5,2}$	81.5	88.2	

Table 7.3 Performances of different classifiers and for different settings

To summarize, training a classifier and testing it with the same data obtained higher precision and recall than train it with one and predict on another (Table 7.4). For example, C4.5 obtained 87.7% precision and 88% recall, which is better than its performance shown in Table 2, to name but a few. CART*, in particular, obtained about 94% precision and 96% recall when trained and tested with the same data. This probably means an overfitted model.

The prediction capability of the trained classifiers on new data is shown in Table 7.3. It shows that most classifiers worked satisfactorily for both settings and outperformed the weighted average approach (Table 7.5), expect for CART*. In general, decision trees provide more interpretable results (i.e. rules) than numerical learning. In addition, Table 7.3 confirms that higher performance in classifying new data is correlated to relatively smaller sizes of generated trees. Among other decision trees, C4.5 appears to be the most promising in this matching task due to its better performance, its stability in reversing training and test sample and its more tractable tree sizes. CART* performed poorly because it over fitted the training samples (see also our discussion in the previous paragraph) and generated over complicated trees, which not only makes the resulting rules more difficult to interpret but also reduces their performance in classifying novel patterns. Concerning C4.5,

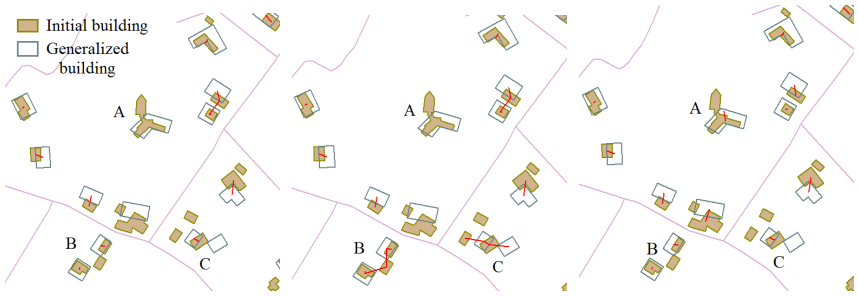
	Setting 3: classifier trained with and tested on A		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	94.3	96.5	242
CART$^{95\%}$	95.1	89.4	128
CART$^{85\%}$	87.1	89.8	45
C4.5	87.7	88.0	19
NB	85.3	86.0	
SVM$^{0.5,2}$	86.2	84.5	
	Setting 4: classifier trained with and tested on B		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	94.8	96.0	325
CART$^{95\%}$	95.2	88.6	186
CART$^{85\%}$	88.2	88.7	94
C4.5	88.4	80.6	11
NB	81.3	84.7	
SVM$^{0.5,2}$	82.3	84.6	
	Setting 5: training data A with 10-folds cross-validation		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	82.6	82.4	242
CART$^{95\%}$	87.8	81.1	128
CART$^{85\%}$	84.7	86.1	45
C4.5	86.3	84.2	19
NB	85.1	85.7	
SVM$^{0.5,2}$	86.1	84.3	
	Setting 6: training data B with 10-folds cross-validation		
Classifier	Precision [%]	Recall [%]	Tree size [leaf no.]
CART*	78.6	80.3	325
CART$^{95\%}$	84.4	79.0	186
CART$^{85\%}$	83.1	82.9	94
C4.5	83.3	82.5	11
NB	81.1	84.7	
SVM$^{0.5,2}$	82.2	84.4	

Table 7.4 Performances of different classifiers and for different settings (continued)

NB and SVM (with $C = 0.5, \gamma = 2$ automatically computed for training samples), no persistent conclusion can be drawn as to the difference in their performance. It is however known that classification accuracy of NB classifier can be further improved (Section 7.4.3).

	Precision	Recall
Weighted average	61.70%	61.70%

Table 7.5 Performance of weighted average approach on study area A with normalized and equally weighted measures



(a) C4.5 (b) Naive Bayes (c) SVM$^{0.5,2}$

Figure 7.12 Matching examples predicted for sample set A by training from set B (links are visualized in red)

Besides, higher precisions and lower recalls can be observed for the classifiers trained with dataset A (characterized mainly by towns) and tested with dataset B (characterized by a mixture of towns, suburb and rural settlements) compared with the reverse setting. Note that both training sets were carefully prepared to gain the same positive class rate (Table 7.2). This ensures that such a difference was not caused by different positive class rates of the training samples. This result suggests that spatial characteristics of data have an impact on classification performance. However, how different characteristics may affect the matching accuracy needs to be further explored.

7.4.3 *Improved matching with domain knowledge*

A closer look at the matching results gets insight into where and why misclassifications occurred. Several typical poor matching was identified:

– Some pairs of buildings (should be matched) were unmatched because their shapes are drastically different given that they are very close (e.g. case A in Fig. 7.12(a) and 7.12(b));

ID(d_i)	ID(g_j)	matched	unmatched
8682	2985	0.324	0.676
10033	2985	0	1

(a)

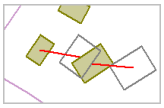
ID(d_i)	ID(g_j)	matched	unmatched
10450	2984	0.68	0.32
10450	2989	0.505	0.495

(b)

Figure 7.13 Probability table for case A (a) and B (b)

ID(d_i)	ID(g_j)	matched	unmatched
10805	3007	0.713	0.287
10805	3008	0.678	0.322

(a)



(b)

Figure 7.14 Probability table for case C (a) and improved situation after incorporating domain knowledge (b)

- Some pairs (should not matched) were mismatched because they are too similar in term of shape, orientation and position (e.g. case B in Fig. 7.12(b));

- Incorrect matching between groups of buildings where n-to-m relationships are likely to happen (e.g. case C in Fig. 7.12(a)– 7.12(c)); in this case, contextual information should improve the matching.

Some of the above misclassifications can be improved by incorporating the generalization knowledge as described in Section 6.2.4. For example, the poor situations A, B, C in Fig. 7.12(b)) by Naive Bayes are improved in the following way.

Case A. Two related pairs are searched in the probability table as shown in Fig. 7.13(a); the generalized object (#2985), with two potential links that were labeled by NB as 'UnMatched'. According to Rule 1 (Section 6.2.4), a link with relatively higher positive probability is selected as best fitted link. The selected building (#8682) proves to be the correct correspondence.

Case B. One initial building (#10450) has two correspondences in generalized dataset, which violates Rule 2 (Section 6.2.4). Since no singularity is found in this cluster, Rule 2 can be applied directly by removing one of the links and the most probable link is selected (Fig. 7.13(b)). The selected correspondence (#2984) is the upper one in the cluster B in Fig. 7.12(b), which is a more reasonable result.

Case C. It is more complex as a singularity is detected (the right most one in cluster C in Fig. 7.12(b)). The detected singularity (#3008) links to the building (#10805) in initial dataset (Fig. 7.14(a)), therefore this link has to be kept. Meanwhile, the other outgoing link from #10805 should be removed according to Rule 2, though it appears to be a more probable link for #10805. After this, the matching result is as follows (Fig. 7.14(b)), and surprisingly this is exactly what the manual matching was like, even without the use of contextual information.

The pair-wise matching allows for n-to-1 and n-to-m relationship to be implicitly modeled (e.g. {{a1,b1}, {a2,b1}, {a2,b2}, {a3,b2}} forms a 3-to-2 relationship). However, current use of domain knowledge (Rule 2) as a post-process is to detect and remove incorrect relationship such as the group C in Fig. 7.12(b), which naturally disallows n-to-m relationships (although n-to-1 is still allowed). Better rules are required to replace Rule 2 in order to distinguish incorrect correspondence and potential n-to-m relationships. A prior matching of building groups as described in [239] may be helpful.

In summary, classifiers with probability structures and soft decisions are more promising in the matching as domain knowledge can be incorporated to improve the performance. As described above, the matching results obtained from Naive Bayes can be improved by further analyzing the probabilities using the domain knowledge (Section 6.2.4). Traditional SVM as used here only provides crisp decisions. However, if a probabilistic SVM [150] is used, the domain knowledge can also be incorporated to improve the SVM-based matching.

7.4.4 *Reflection on the matching criteria*

Section 6.2 presents a first attempt into the classification-based approach to data matching where multivariate decision is important. So the selection of optimal criteria (and measures) to achieve the best possible matching results was not the focus. Four categories of criteria (position, size, shape and orientation) were used based on commonsense knowledge. A correlation analysis (similar to the one used in [43, 224]) was later carried out which shows no significant correlation between the four measures. However, one should note that the categories are by no means complete and the measures used to evaluate the criteria may not be the optimal ones.

For example, it is questionable whether to use the size ratio for the matching because different size ratios can be caused by enlarging smaller objects, though a distribution of size ratios can be learnt which may facilitate the

classification. To get more insights, we carried out parallel experiments where the size criterion was removed. For setting 1 we found that for C4.5, NB and SVM precision decreases and recall increases, indicating that while more true positives (correct links) were found, even more false positives were also produced, which is arguably undesirable. For CART of different versions both precision and recall decrease. Similar results were obtained for setting 2. This suggests that the size criterion adds more or less to the matching. However, a redesign of size criterion in the future taking into account the possible change ratio in relation to initial sizes may give more discriminating power.

Likewise, by removing shape respective orientation criteria, obvious decrease in both precision and recall occurs for the classifiers. This suggests that the used measures are relevant for building matching though better performance can be anticipated by designing measures that differentiate special cases (e.g. oval shapes).

In future research, more measures should be analyzed for different criterion categories, the optimal one or combination of ones can be chosen using techniques such as principal component analysis [43]. Further, other criterion categories such as semantic and contextual ones can be integrated to improve the data matching.

7.5 Preservation of building alignments

This section presents the results of automated evaluation of preservation of building alignments. After outlining the design and assumption of the experiments (Section 7.5.1), we evaluate the three constraints described in Section 6.4.1 in an automatic manner after building alignments being recognized (Section 7.5.2) and corresponding alignments being linked (Section 7.5.3). The evaluation approach was implemented in DoMap. The results are presented and discussed in Sections 7.5.4–7.5.7.

7.5.1 *Experiment design and assumption*

To validate the proposed evaluation approach (Section 6.4), it was applied to evaluate topographic datasets of Kadaster, the Netherlands (Fig. 7.15) against constraints on alignments. TOP10NL is the initial dataset and several datasets (at 1:50k) derived from TOP10NL are the target datasets. Note that we focus on the regions where conceptualization has not changed, meaning that those initial buildings that were transformed into built-up areas due to their high densities are not considered in the evaluation process.

(a) TOP10NL (b) TOP50NL (c) Output1 (d) Output2

Figure 7.15 Excerpts of the tested datasets used for evaluating the preservation of alignments (Data source: Kadaster, NL)

TOP50NL is interactively (manually) generalized (by cartographers) from TOP10NL and serves as a good example. By 'good generalization' we mean that in focused regions the alignments are well generalized, i.e., related constraints are respected as much as possible. Output1 and Output2 are automated solutions with or without a consideration of alignment structures. Visually compared with TOP50NL, Output1 is assumed as an acceptable solution and Output2 as a bad example. In fact, buildings in Output2 were selected solely based on sizes and hence most alignments were deleted.

The assumption of the experiment design is that the evaluation approach should obtain higher scores (i.e. lower violations) for good examples and lower scores (i.e. higher violations) for bad examples. The final evaluation results should also reflect the differences in the generalization of building alignments as described above.

7.5.2 *Alignments recognized*

The recognition approach described in Section 5.4 was applied to the initial and target datasets in Fig. 7.15 with the same parameter values. Recognition results are exemplified in Fig. 7.16.

7.5.3 *Linking corresponding building alignments*

The *ad-hoc* matching algorithm described in Section 6.3.3 was implemented in DoMap and the links are stored in a relational database per dataset. The matching results are exemplified in Fig. 7.16. A visual inspection of the matching results show that the *ad-hoc* matching approach gave 100% correct

Figure 7.16 Constraint violations are visualized for TOP50 (a) and Output1 (b): target alignments are matched to and evaluated against initial ones concerning the constraint on spatial distribution

	Number of alignments		
Dataset	Matched	Total	Changed to 1:50k built-up areas
TOP10NL	50	147	78
TOP50NL	36	44	
TOP10NL	38	147	87
Output1	38	55	
TOP10NL	2	147	0
Output2	2	3	

Table 7.6 Matching of recognized alignments – a summary

matching for our test datasets. It shows also that partial matching between initial and target alignments is rather common. In addition, matched and unmatched building alignments for the test datasets (Fig. 7.15) are summarized in Table 7.6.

7.5.4 *Evaluation results*

This section presents the evaluation results for the tested datasets. Evaluation of individual alignments have been demonstrated in Section 6.4.2, which is useful for conflict detecting and evaluation-for-editing [see e.g. 120]. The thresholds and tolerances used in the evaluation process can be specified by users. In our experiment, we used $T_1 = 2.5°$, $T_2 = 25°$

	WACV of the three constraints			
	Existence	Existence†	Orientation	Distribution
TOP50NL	0.68	0.28	0.11	0.04
Output1	0.70	0.37	0.07	0.23
Output2	0.99	0.99	0.05	0

† This is calculated by deducing those alignments that are transformed into built-up areas.

Table 7.7 Evaluation results aggregated for the three 1:50k datasets

for the alignment orientation constraint and $T_3 = -0.015$, $T_4 = -0.15$ for the distribution of composing object constraint. The evaluation results are summarized in Table 7.7 to rank the quality of the generalization outputs concerning these constraints. Note that constraint violation to alignment existence can be simply calculated by the values in Table 7.7, so only individual assessments on alignment orientation and spatial distribution of composing buildings are presented in Tables 7.8–7.11. Since Output2 contains only three alignments and two of which were correctly matched to TOP10NL, we do not show the details.

7.5.5 *Appropriateness of the approach and the constraints*

Although the result in Table 7.7 appears that all the 1:50k datasets largely violate the constraint on the existence of alignments. The high violations are partly because that many alignments were transformed to built-up areas to meet another important Dutch specification 'groups of buildings with densities >= 0.1 should be changed into built-up areas'. Since we do not consider these alignments, the violation degree is recalculated by leaving out these alignments. For example, 78 out of 147 initial (TOP10) alignments were transformed to TOP50NL built-up areas (Table 7.6), so 69 initial alignments are left in which 50 are kept in TOP50NL. In view of this, the new violation of this constraint becomes around 0.28 for TOP50NL (see the third column in Table 7.7). Similarly, a violation of around 0.37 can be calculated for Output1. Since there is no alignment that was transformed to built-up areas in Output2, the violation remains 0.99 indicating that almost all the alignments were deleted (only two alignments are kept).

The above results correspond exactly to the observations made in Section 7.5.1. That is, alignments in TOP50NL and Output1 are better generalized, where TOP50NL is even better. The alignments in Output2 are badly generalized. Relatively better results (lower CVs) are obtained for the two

Table 7.8 Evaluation result of TOP50NL dataset on 'alignment orientation' ($T_1 = 2.5°$, $T_2 = 25°$)

InitAlignID	TargAlignID	InitType	TargetType	ODev (°)	CV	CI
167	1	Curvilinear	Collinear	40.45	1	0.75
168	1	Curvilinear	Collinear	4.39	0.08	0.75
167	2	Curvilinear	Collinear	56.24	1	0.5
108	3	Collinear	Collinear	0.45	0	1
109	3	Collinear	Collinear	6.86	0.19	1
149	5	Collinear	Collinear	1.82	0	1
83	6	Collinear	Collinear	0.33	0	1
57	7	Collinear	Collinear	8.12	0.25	1
151	8	Collinear	Collinear	3.7	0.05	1
152	8	Collinear	Collinear	6.69	0.19	1
156	9	Collinear	Collinear	1.92	0	0.8
103	10	Collinear	Collinear	0.09	0	1
86	12	Collinear	Collinear	8.24	0.26	1
85	13	Collinear	Collinear	1.9	0	1
79	14	Collinear	Collinear	2.91	0.02	0.8333
160	19	Collinear	Collinear	1.46	0	1
118	20	Collinear	Collinear	8.78	0.28	1
146	21	Collinear	Collinear	2.33	0	0.8
73	22	Collinear	Collinear	4.52	0.09	1
179	23	Curvilinear	Collinear	1.58	0	1
87	24	Collinear	Collinear	1.04	0	1
48	25	Collinear	Curvilinear	3.19	0.03	0.6667
166	25	Curvilinear	Curvilinear	7.64	0.23	1
107	27	Collinear	Curvilinear	1	0	1
164	27	Collinear	Curvilinear	11.72	0.41	1
162	28	Collinear	Curvilinear	4.46	0.09	1
163	28	Collinear	Curvilinear	24.55	0.98	1
72	29	Collinear	Curvilinear	3.34	0.04	1
104	30	Collinear	Curvilinear	15.29	0.57	1
105	31	Collinear	Curvilinear	2.29	0	0.8
106	31	Collinear	Curvilinear	2.91	0.02	1
190	31	Curvilinear	Curvilinear	4.36	0.08	1
175	32	Curvilinear	Curvilinear	0.51	0	1
185	33	Curvilinear	Curvilinear	3.07	0.03	1
186	33	Curvilinear	Curvilinear	0.17	0	1
187	33	Collinear	Curvilinear	2.5	0	1
188	33	Collinear	Curvilinear	22.28	0.88	1
120	34	Collinear	Curvilinear	0.01	0	1
189	35	Curvilinear	Curvilinear	4.6	0.09	0.75
147	36	Collinear	Curvilinear	3.46	0.04	1
75	37	Collinear	Curvilinear	2.26	0	1
171	38	Curvilinear	Curvilinear	0.52	0	1
63	39	Collinear	Curvilinear	2.78	0.01	1
178	39	Curvilinear	Curvilinear	4.47	0.09	1
49	40	Collinear	Curvilinear	0.17	0	1
153	42	Collinear	Curvilinear	1.18	0	1
154	42	Collinear	Curvilinear	4.58	0.09	1
155	42	Collinear	Curvilinear	4.23	0.08	1
132	43	Collinear	Curvilinear	3.08	0.03	1
133	43	Collinear	Curvilinear	2.48	0	0.8571
182	44	Curvilinear	Collinear	3.59	0.05	1
		*Average*CV = 0.14				*WACV* = 0.11

Table 7.9 Evaluation result of TOP50NL dataset on 'spatial distribution of composing buildings' (T_3 = -0.015, T_4 = -0.15)

TargAlignID	InitAlignNo	Initial Homo	Target Homo	HDev	CV	CI
1	2	0.73	0.84	0.11	0	0.75
2	1	0.77	0.82	0.05	0	0.33
3	2	0.7033	0.91	0.2067	0	0.71
5	1	1	0.98	-0.02	0.04	1
6	1	0.83	0.85	0.02	0	0.67
7	1	0.95	0.9	-0.05	0.26	0.75
8	2	0.8	0.79	-0.01	0	0.86
9	1	0.85	0.96	0.11	0	0.8
10	1	0.92	0.98	0.06	0	1
12	1	0.83	0.92	0.09	0	1
13	1	0.78	0.9	0.12	0	1
14	1	0.81	0.87	0.06	0	0.5
19	1	0.66	0.79	0.13	0	1
20	1	0.96	0.98	0.02	0	0.67
21	1	0.83	0.98	0.15	0	0.8
22	1	0.93	0.86	-0.07	0.41	0.5
23	1	0.68	0.95	0.27	0	1
24	1	0.86	0.91	0.05	0	0.5
25	2	0.8243	0.85	0.0257	0	0.69
27	2	0.8314	0.94	0.1086	0	1
28	2	0.7843	0.87	0.0857	0	0.83
29	1	0.82	0.96	0.14	0	0.75
30	1	0.93	0.9	-0.03	0.11	0.67
31	3	0.8409	0.9	0.0591	0	0.76
32	1	0.85	0.84	-0.01	0	1
33	4	0.7917	0.82	0.0283	0	1
34	1	0.82	0.91	0.09	0	1
35	1	0.83	0.91	0.08	0	0.56
36	1	0.88	0.89	0.01	0	1
37	1	0.94	0.93	-0.01	0	0.4
38	1	0.75	0.9	0.15	0	1
39	2	0.8471	0.91	0.0629	0	0.75
40	1	0.94	0.87	-0.07	0.41	0.5
42	3	0.8692	0.84	-0.0292	0.11	0.55
43	2	0.823	0.78	-0.043	0.21	0.9
44	1	0.8	0.91	0.11	0	1
		$Average CV = 0.04$				$WACV = 0.04$

Table 7.10 Evaluation result of Output1 dataset on 'alignment orientation' ($T_1 = 2.5°$, $T_2 = 25°$)

InitAlignID	TargetAlignID	InitType	TargetType	ODev (°)	CV	CI
8306	2	Curvilinear	Collinear	0.6	0.00	1
8304	4	Curvilinear	Collinear	1.83	0.00	0.4286
8296	5	Curvilinear	Collinear	0.6	0.00	0.5714
8309	8	Curvilinear	Collinear	4.26	0.08	0.6
8352	9	Collinear	Collinear	0.8	0.00	0.75
8345	10	Collinear	Collinear	1.66	0.00	1
8267	11	Curvilinear	Collinear	15	0.56	1
8283	12	Curvilinear	Collinear	1.14	0.00	0.6667
8281	13	Curvilinear	Collinear	7.43	0.22	1
8323	14	Curvilinear	Collinear	0.33	0.00	1
8268	15	Curvilinear	Curvilinear	6.32	0.17	0.8
8269	15	Curvilinear	Curvilinear	5.87	0.15	1
8267	16	Curvilinear	Curvilinear	15	0.56	1
8262	18	Curvilinear	Collinear	0.37	0.00	1
8248	19	Curvilinear	Collinear	0.43	0.00	0.4286
8249	20	Curvilinear	Collinear	1.3	0.00	1
8354	21	Collinear	Collinear	2.41	0.00	1
8316	23	Curvilinear	Curvilinear	4.69	0.10	1
8315	24	Curvilinear	Collinear	5.76	0.14	0.75
8245	25	Curvilinear	Collinear	7.21	0.21	1
8244	26	Curvilinear	Collinear	3.3	0.04	1
8246	26	Curvilinear	Collinear	1.18	0.00	1
8334	27	Collinear	Collinear	2	0.00	1
8312	28	Curvilinear	Collinear	2.43	0.00	1
8335	29	Collinear	Collinear	9.15	0.30	0.6667
8337	30	Collinear	Collinear	4.42	0.09	1
8237	31	Curvilinear	Collinear	2.35	0.00	1
8336	32	Collinear	Collinear	1.46	0.00	0.8571
8242	33	Curvilinear	Collinear	4.89	0.11	0.5
8339	35	Collinear	Collinear	6.76	0.19	0.3636
8340	37	Collinear	Collinear	6.95	0.20	0.6
8250	39	Curvilinear	Collinear	2.06	0.00	0.5
8250	40	Curvilinear	Collinear	2.63	0.01	0.75
8259	42	Curvilinear	Collinear	1.04	0.00	0.4
8349	43	Collinear	Collinear	3.21	0.03	0.6111
8348	44	Collinear	Collinear	3.25	0.03	1
8222	46	Curvilinear	Collinear	1.11	0.00	0.8333
8221	47	Curvilinear	Curvilinear	0.55	0.00	0.8
8212	54	Curvilinear	Collinear	2.27	0.00	1
8220	55	Curvilinear	Collinear	0.16	0.00	1
		$Average CV = 0.08$				$WACV = 0.07$

Table 7.11 Evaluation result of Output1 dataset on 'spatial distribution of composing buildings' (T_3 = -0.015, T_4 = -0.15)

TargAlignID	NoInitAlign	Initial Homo	Target Homo	HDev	CV	CI
2	1	0.95	0.82	-0.13	0.85	0.67
4	1	0.82	0.94	0.12	0.00	0.17
5	1	0.79	0.99	0.2	0.00	0.57
8	1	0.83	0.96	0.13	0.00	0.6
9	1	0.83	0.83	0	0.00	0.56
10	1	0.8	0.89	0.09	0.00	0.8
11	1	0.93	0.83	-0.1	0.63	0.4
12	1	0.82	0.96	0.14	0.00	0.67
13	1	0.96	0.97	0.01	0.00	0.67
14	1	0.66	0.8	0.14	0.00	1
15	2	0.8112	0.95	0.1388	0.00	0.88
16	1	0.93	0.82	-0.11	0.70	0.67
18	1	0.91	0.91	0	0.00	1
19	1	0.78	0.96	0.18	0.00	0.43
20	1	0.83	0.82	-0.01	0.00	1
21	1	0.86	0.97	0.11	0.00	1
23	1	0.83	0.85	0.02	0.00	0.5
24	1	0.83	0.77	-0.06	0.33	0.45
25	1	0.74	0.88	0.14	0.00	1
26	2	0.8986	0.73	-0.1686	1.00	1
27	1	0.75	0.84	0.09	0.00	1
28	1	1	0.97	-0.03	0.11	1
29	1	0.94	0.94	0	0.00	0.44
30	1	0.9	0.81	-0.09	0.56	0.5
31	1	0.91	0.77	-0.14	0.93	1
32	1	0.76	0.89	0.13	0.00	0.86
33	1	0.81	0.96	0.15	0.00	0.33
35	1	0.71	0.97	0.26	0.00	0.36
37	1	0.75	0.89	0.14	0.00	0.6
39	1	0.86	0.99	0.13	0.00	0.25
40	1	0.86	0.96	0.1	0.00	0.5
42	1	0.76	1	0.24	0.00	0.4
43	1	0.77	0.89	0.12	0.00	0.61
44	1	0.87	0.93	0.06	0.00	0.5
46	1	0.96	0.93	-0.03	0.11	0.63
47	1	0.93	0.85	-0.08	0.48	0.6
54	1	0.94	0.79	-0.15	1.00	0.38
55	1	0.95	0.79	-0.16	1.00	0.5
		$Average CV = 0.20$				$WACV = 0.23$

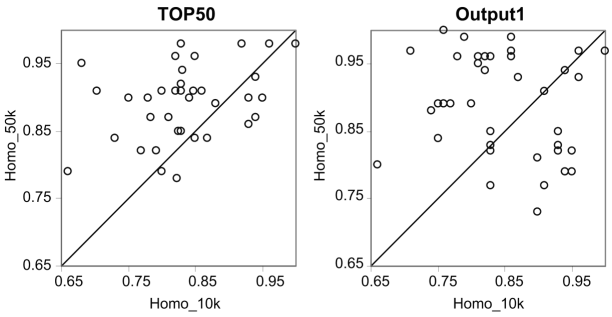


Figure 7.17 Plot of initial and target homogeneities (each point represents a pair of corresponding alignments) for TOP50NL (left) and Output1 (right)

property-level constraints. This is because the constraints only consider the alignments that were represented in target data, and no abrupt changes to the relative positions of alignment buildings were introduced to the target datasets. In terms of the spatial distribution of composing buildings, TOP50 was generalized much better than Output1 (see also Fig. 7.16).

To sum up, the evaluation approach was successful in the sense that it gave higher scores to the generalized datasets with better cartographic quality and lower scores to those with lower quality (the quality of generalized alignments). Note that over-optimistic results can be caused if the evaluation relies solely on the two property-level (orientation and distribution) constraints. To avoid over-optimistic results, we suggest also using the entity-level (existence) constraint. and thereby the entity-level (existence) constraint has to be used to avoid over-optimistic results. In this respect, the constraint on alignment existence is more relevant to the textbook requirement [184]: "alignments should be represented as such at smaller scales". The constraints on alignment orientation and distribution provide additional information as to how properties of the alignments should be preserved.

7.5.6 *Need for more formal constraints*

The evaluation result of the constraint on the spatial distribution of composing buildings (Fig. 7.17) gives more insight into how building alignments behave at scale transitions. In both generalized datasets, especially TOP50NL, most 1:50k alignments tended to become more regular rather than just preserve the regularity (i.e. homogeneity values are above the preservation line). A similar requirement is documented in generalization textbooks like SSC [184]: "with decreasing scale irregular spacing disappears [...]". This also explains why there are always new alignments (0-to-1 relationship) emerging from groups of buildings that are irregularly distributed in the

initial dataset. At present, this requirement is not in the NMA specifications and hence should be added for automated processes to obtain expected results. The requirement can be formally expressed as:

$$\begin{aligned}&\left(\forall al_i^I\right)\left(\exists al_i^T\right)\left(Corr\left(al_i^I, al_i^T\right)\right)\\&\Rightarrow\left(Homo\left(al_i^T\right) = K \times Homo\left(al_i^I\right) \pm \sigma\right)\end{aligned} \tag{7.1}$$

where $K > 1$; σ is a user defined tolerance. To summarize, this evaluation approach presents a precise model for the three particular constraints on alignments, where the concept and measures of the constraints and their allowed changes are mathematically defined. These should be added to enrich to the specifications for automated processes (see e.g. C6–C8 in Section 4.3). Further formalization can be achieved using formal languages such as Object Constraint Language (OCL) as shown in Stoter et al. [194], where spatial data model and inter-scale semantics (e.g. map specifications) can be formally defined in UML and OCL constraints.

7.5.7 *Limitations*

The uncertainty issue caused by the partial matching of alignments cannot be avoided within the three-step evaluation approach. The issue is currently addressed by introducing a confidence indicator to each assessment, given the fact that the partial matching was not very critical in our experiment. For example less than 20% of them were partially matched between TOP10NL and TOP50NL. For the partially matched alignments, the average match degree is 0.75. Still, it is possible to reduce (part of) the uncertainty in the future (Section 8.3.2).

7.6 Preservation of spatial distribution

This section validates the density and distance based approach to evaluating the preservation of spatial distributions (Sections 7.6.1 and 7.6.2). The advantages and limitations of the two approaches with respect to each other are discussed in Section 7.6.3.

7.6.1 *Density-based approach*

We applied the density-based measure of spatial distribution (Section 6.5) to road and building datasets respectively (Fig. 7.18 and 7.19). First, the corresponding relationship between spatial objects in the initial and target datasets should be created. We did not establish the links using the generic

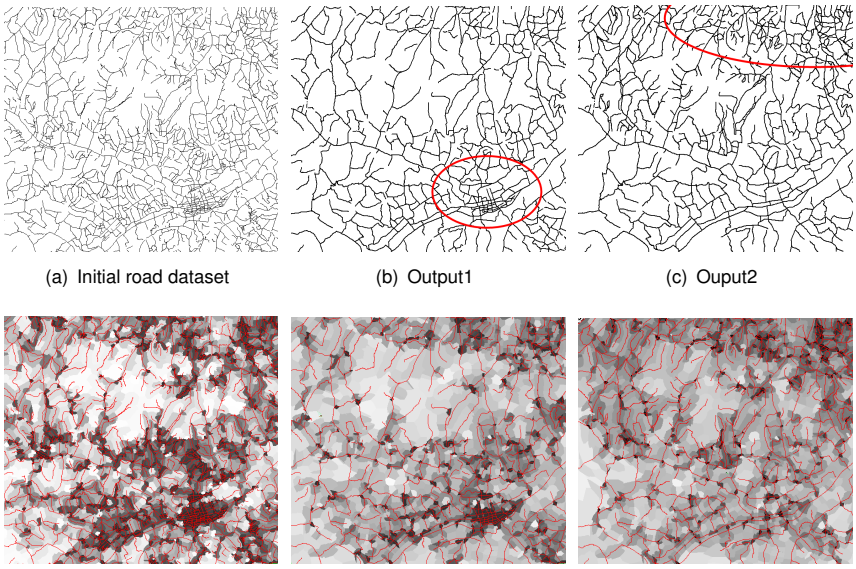
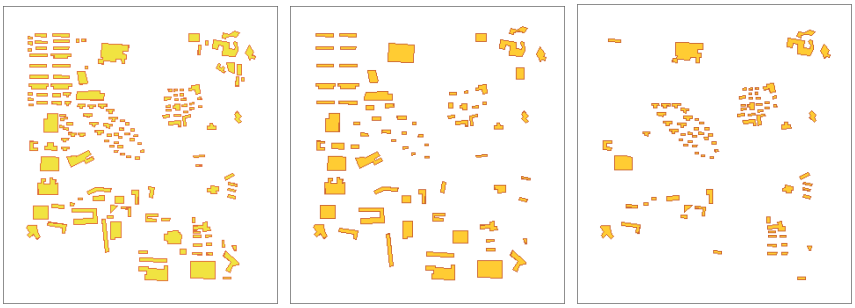


Figure 7.18 Initial roads at 1:10k and two generalization outputs at 1:50k (b) and (c) with with local densities enriched (d)–(f) (Data source: Jilin, China)

data matching approach described in Section 6.2, because the matching accuracy is not yet adequate for real applications (see Section 7.4.2). Instead, we created the links during the generalization process.

We firstly investigate how the spatial distribution of road features can be evaluated. The initial road in Fig. 7.18(a) was interactively generalized in two different ways. Output1 (Fig. 7.18(b)) is a better solution than Output2 (Fig. 7.18(c)) since in Output1 the balance of densities was better preserved than in Output2. A case in point is that the town center with higher densities (highlighted by the red circle in Fig. 7.18(b)) was kept in Output1. Besides, the density difference was also preserved in Output1, which was violated in Output2 (densely populated area is circled in red in Fig. 7.18(c); the town center is no longer recognizable). The differences between Output1 and Output2 with respect to the initial data is also apparent in density maps (Fig. 7.18(d)–(f)).

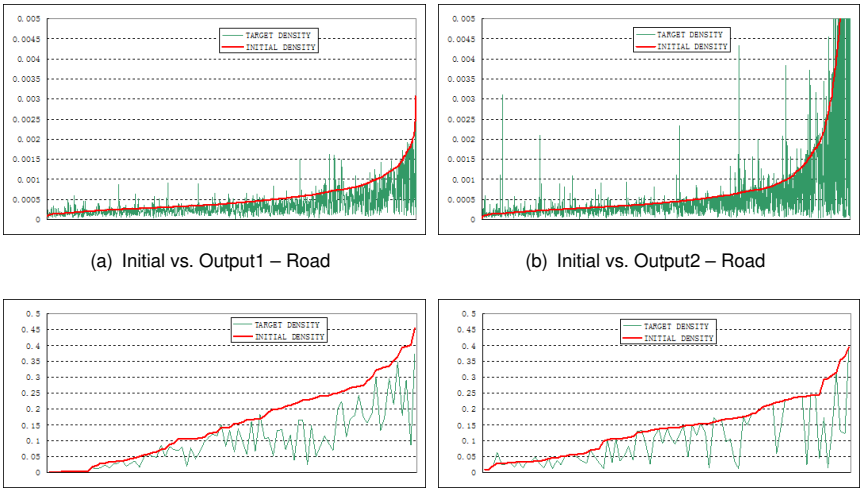
Likewise, two building datasets(Output1 and Output2 in Fig. 7.19(b) and 7.19(c)) were derived from the initial dataset (see Fig. 7.19) via interactive generalization. Output1 is a better generalization than Output2 in terms of



(a) Initial building dataset (b) Output1 (c) Output2

Figure 7.19 Initial buildings (a) at 1:10k and two generalization outputs at 1:50k (b) and (c) (Data source: Shenzhen, China)

spatial distribution. Numerical analysis of the generalization of the initial road and building datasets are based on the methods in Section 6.5 and the results are presented as follows.



(a) Initial vs. Output1 – Road (b) Initial vs. Output2 – Road

(c) Initial vs. Output1 – Building (d) Initial vs. Output2 – Building

Figure 7.20 Relating initial (red) and target (green) local densities

To begin with, local densities of corresponding entities are displayed in Fig. 7.20 (aggregation into meso-densities in case of n-to-m relationships is necessary). As can be seen from the figure, the generalized densities

	Correlation	MeanDCR	StdDCR	CVarDCR
Road Output1	0.69	0.71	0.39	0.55
Road Output2	0.55	0.83	0.72	0.88
Building Output1	0.83	0.71	0.28	0.39
Building Output2	0.71	0.76	0.35	0.46

Table 7.12 Numerical analysis of change of local densities

are in general lower than the initial densities. This is mainly because the generalization did not take symbols into account.

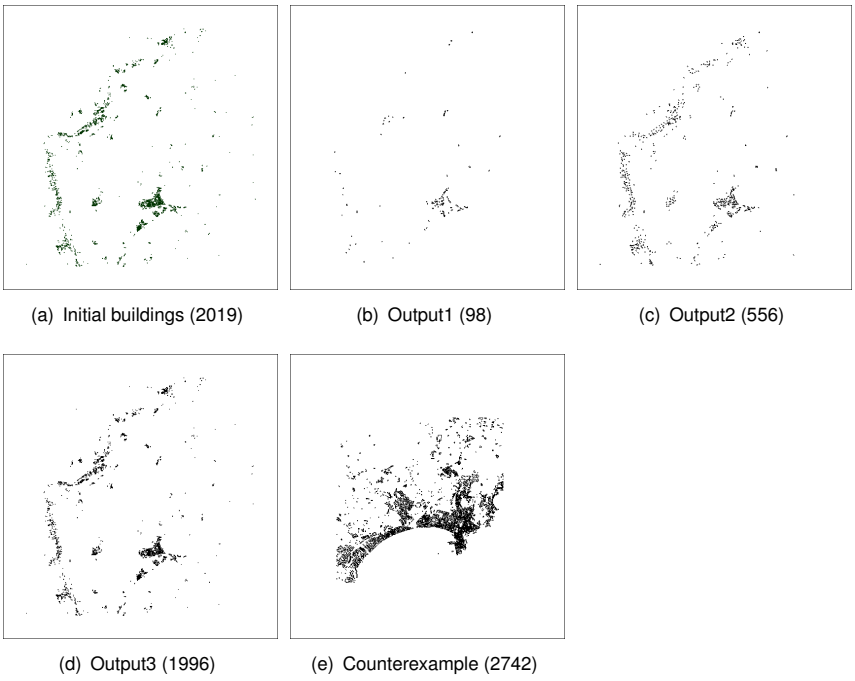
To evaluate the preservation of spatial distribution (similarity), correlation between target and initial densities is analyzed. In addition, density change ratios (DCR = Target density / Initial density) are calculated for each pair of corresponding entities, from which mean ratio (MeanDCR), standard deviation of DCR (StdDCR), and coefficient of variation of DCR (CVarDCR) are calculated to assist the analysis (Table 7.12).

The numerical analysis confirms that, for both road and building datasets Output1 is more similar to initial data than Output2 in terms of spatial distribution, as both Output1's are more correlated to the initial data. The analysis also reveals that the mean density change ratios for Output1 and Output2 for both road and building data are more or less the same. But the variance (StdDCR and CVarDCR) of Output1 is smaller than that of Output2 for both road and building data. This indicates that DCR is more persistent for Output1 than for Output2, implying also that Output1 is more similar to the initial data than Output2 for both road and building data.

Note that, correlation ranges from -1 to $+1$. It is therefore easier to convert correlation into constraint satisfaction (or violation). For example, when correlation equals to 1, the two distributions are 100% correlated and hence the constraint on keeping spatial distribution similarity is completely satisfied ($CS = 1$). However, completely correlated distributions are not necessarily identical.

7.6.2 *Distance-based approach*

This section presents the implementation and validation of the distance-based measure of similarity between spatial distributions described in Section 5.6. This approach is easier to implement, requiring only nearest distance calculation. This approach was tested against both real topographic



(a) Initial buildings (2019) (b) Output1 (98) (c) Output2 (556)

(d) Output3 (1996) (e) Counterexample (2742)

Figure 7.21 Topographic datasets: (a)–(d) the initial data and its generalization outputs (No. of buildings) (Data source: IGN, france) and (e) a counterexample (Data source: ICC, Catalonia, Spain)

data and synthetical data.

Validation by topographic data

As shown in Fig. 7.21, the topographic data used is the IGN datasets resulting from the EuroSDR project [190, 192], where an ICC dataset is used as a counterexample. To calculate the distance-based similarity measure, the ICC dataset was displaced to the position that are more or less collocated to the initial dataset (i.e. the extent of the ICC dataset is approximately equal to that of the initial dataset).

Note that to calculate nearest inter-set distance between distributions of building polygons, we treated these polygons as points (using their centroids). To give an intuition of the similarity between spatial distributions, NDS_{AB} and NDS_{BA} (Section 5.6.1) are calculated for the datasets in Fig. 7.21 and are plotted in Fig. 7.22.

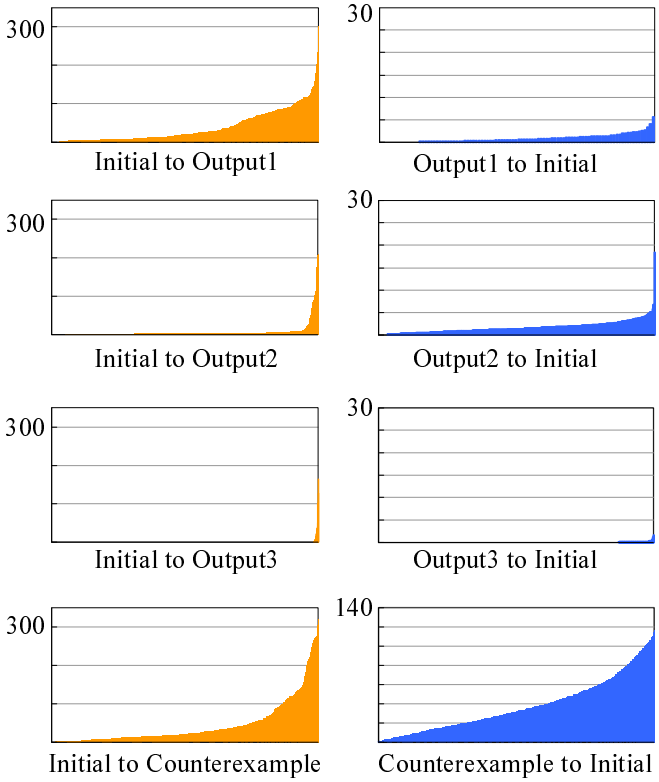


Figure 7.22 Plots of NDS_{AB} (left column) and NDS_{BA} (right column) for the topographic datasets (Let A be the initial data and B be the target data)

Several observations can be made in Fig. 7.22. For the outputs that properly preserve the spatial distribution (Output2 and Output3), more than 90% of the NDs from the initial data to generalized outputs are relatively low (lower than 10mm – map unit at 1:10k); only a small portion of NDs is relatively high, with very few exceptions of abnormally large distances (100–200mm). Those extremely high distances are caused by removing the isolated small buildings located in remote areas (such as rurual and mountain areas) from the initial dataset. A closer look at the NDs from Output2 and Output3 to the initial dataset reveals that almost all the NDs are below 5mm. Although the NDs from Output1 to the initial data are also low, the NDs from the initial data to output1 are significantly large. This indicates that Output1 is less similar to the initial data than Output2 and Output3.

In order to show that the similarity can also identify spatial distributions that are dissimilar, an independent dataset was used (the counterexample

Between Initial and [...]		Output1	Output2	Output3	Counter
$NISD(A,B)$		37.87	7.33	0.48	43.92
$NISD(B,A)$		0.84	2.13	0.04	37.69
$\|\|Sim(\cdot)\|\|$	L^2-norm	37.88	7.63	0.48	57.87
	L^∞-norm	37.87	7.33	0.48	43.92

Table 7.13 Similarity values (non-normalized) between the topographic data sets; $NISD(\cdot)$ is based on weighted mean function F_1 (unit: millimeter at 1:10k)

in Fig. 7.21 which is deliberately displaced to more or less the same location of the initial dataset). The plotting of NDs shows that the NDs from both directions (A to B and B to A) consist of many large distances (e.g. > 20mm) which are much larger than in the other pairs of comparisons in Fig. 7.22. This means that in general the visualization of NDS_{AB} and NDS_{BA} can intuitively describe the similarity and dissimilarity between spatial distributions.

Table 7.13 shows that the degrees of similarity between the target data to the initial data in an ascendent order is: Counterexample $\prec$ Output1 $\prec$ Output2 $\prec$ Output3, where Output3 is the most similar to the initial data. A common finding is that $NISD(B,A)$ were very small and also smaller than $NISD(A,B)$. This is because generalization output can be viewed as subsets of the initial data with some degrees of displacement. Additionally, it seems that the more buildings are selected, the more the output is similar to the initial building distribution. This is however not a sufficient and necessary relation, and we will study the relationship between selection ratio and the similarity using artificial data (see the next section). Finally, the comparison between the initial building and the independent dataset (i.e. the counterexample) indicates that for dissimilar distributions, both $NISD(A,B)$ and $NISD(A,B)$ were quite large.

Validation by artificial data

The observations made in the previous section using topographic data is incomplete, especially the finding of the relationship between selection ratio and degree of similarity concerning spatial distribution. The finding was made based on a limited set of test datasets and may not necessarily be true. According to the theoretical analysis in Section 5.6.2, even if the selection ratio is exactly the same the similarity degrees may be different. In this section, we will demonstrate this point using artificial data sets (Fig. 7.23).

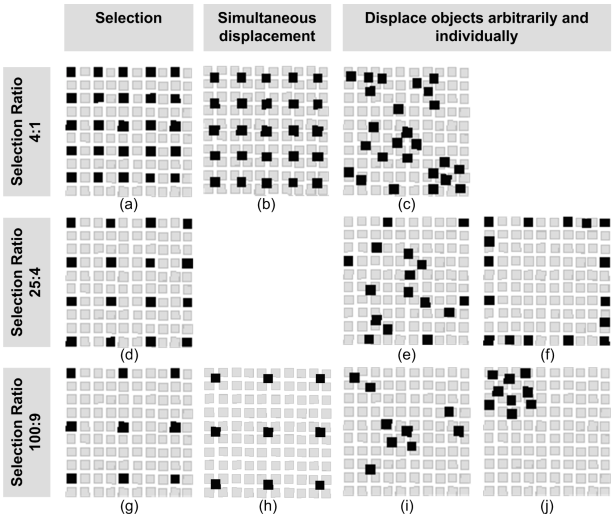


Figure 7.23 Artificial data sets used to demonstrate the expressive power of the similarity measure (gray: initial distribution; black: target distribution)

In Fig. 7.23, all target datasets (black) are superimposed on the initial dataset (gray); the initial dataset consists of 100 artificial buildings. Note that these buildings in the initial data set were generated at 1:10k, meaning that the sizes and spacing of these buildings take into account their cartographic representations at 1:10k. For simplicity, these objects were arranged into a regular (grid-like) pattern which may not exist in real data. The three rows represent different selection ratios. The different columns represent how the generalized objects were transformed into different distributions. In the first column, objects were selected on a regular basis from the initial data without any other operators. Then they were displaced simultaneously to better fit the initial distribution as much as possible (the second column). The last two columns are solutions where these objects were displaced individually and randomly. The degrees of similarity are calculated and presented in Table 7.14.

Table 7.14 shows that in general the similarity between two building distributions increases (the lower the value the more similar two distributions are) with selection ratio. If we look at the solutions with the same selection ratio, the proposed method was able to distinguish between different spatial distributions and the results are consistent with our perception. Specifically, the results suggest that comparison pairs in the last two columns of Fig. 7.23 are less similar than the pairs in the first two columns of the same selection ratio. Our perception tells that the outputs in the second column of Fig. 7.23 (if applicable) are better than the first column. Such a difference can be identified by the similarity measure with uniform norm ($\|Sim(\cdot)\|_\infty$)

Selection Ratio	4:1			25:1			100:9			
Test Case	a	b	c	d	e	f	g	h	i	j
$NISD(A,B)$	2.38	2.01	2.83	2.80	3.38	4.23	4.05	3.86	5.07	10.69
$NISD(B,A)$	0	1.90	1.38	0	0.37	0.32	0	1.78	1.32	1.31
$\|\|Sim(\cdot)\|\|_2$	2.38	2.77	3.15	2.80	3.46	4.24	4.05	4.25	5.24	10.77
$\|\|Sim(\cdot)\|\|_\infty$	2.38	2.01	2.83	2.80	3.38	4.23	4.05	3.86	5.07	10.69

Table 7.14 Similarity values (non-normalized) between the artificial data sets; $NISD(\cdot)$ is based on weighted mean function F_1; unit of the values are millimeter at 1:10k

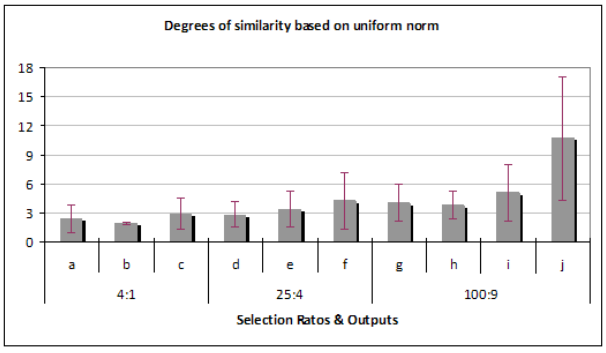


Figure 7.24 Visualization of the degrees of similarity for the artificial data based on uniform norm ($\|Sim(\cdot)\|_\infty$)

but not the measure with Euclidean norm ($\|Sim(\cdot)\|_2$). The measure with uniform norm yielded the lowest values for (b) and (h), see the gray cells in Table 7.14. The degree of similarity based on the uniform norm is visualized in Fig. 7.24. It is also evident in this figure that the test cases (b), (d), and (h) are the optimal solution (standard deviations are also the lowest) among all the test cases under the same selection ratio. The measured similarity degrees confirm our visual perception as to the similarity between the artificial datasets in terms of spatial distribution.

7.6.3 *Limitations*

In general, both the density-based and distance-based measure of similarity between spatial distributions are able to give correct predications as to which two distributions are similar and dissimilar. In spite of spatial distribution, the density-based approach is also able to evaluate density related map specifications. The major merit of the density-based approach is that it precisely define local densities such that the similarity measure takes into account all metric information; whereas the distance-based approach approximate the situation in the case of areal features by abstracting them

as centroids. In addition, the density-based approach is able to compare the similarity between spatial distributions of linear features such as networks, while the latter approach deals only with point-like features. We will discuss further improvement to the distance-based approach in Section 8.3.2.

The density-based approach has several limitations. First, it requires larger amount of computation than the distance-based approach. That is, it first calculates Delaunay triangulation and the Voronoi-like diagram (both of which are computational intensive, see Appendix A); then the similarity measure requires the linkage between corresponding individuals be maintained for the automated evaluation. In this respect, the distance-based approach requires only simple computation, i.e., distance measure, and is thus much easier to implement and practically more valuable. Second, the similarity measure based on densities starts from objects in the target dataset and hence densities of those removed objects are not evaluated. This could be problematic and deserves further research (see Section 8.3.2). The distance-based approach conversely compare the similarity (*NISD*) from both directions and hence should give a complete view of the similarity between the two distributions. Third, one should be more careful in interpreting the evaluation results from the density-based approach. The similarity is currently measured using correlation coefficient. A completely correlated value (Correlation = 1) does not mean that the two distributions are identical but only that the two are similar. For example, the list of densities {1, 2, 3, 4, 5} is completely correlated to {2, 4, 6, 8, 10} as well as to {3, 6, 9, 12, 15}. This means that the correlation measure cannot relate the change of densities to scale reduction. To address this issue, a density change ratio (DCR) and its variance are useful and should be specified for the generalization of a given scale transition (e.g. from 1:10k to 1:50k). This part of information should be further formalized into formal map specifications for practical uses.

8 Conclusions and recommendations

The main objective of this thesis is *to answer the question as to how to evaluate the quality of generalized general-purpose topographic data in a (semi-)automated way, so that a meaningful evaluation can be reached.*

To clarify the scene, this thesis has defined general-purpose generalization of topographic data as a fundamental generalization mode driven by NMA map specifications and distinguished it from task dependent (on-demand) generalization (Section 1.1.2). The two generalization modes are closely related because the former imposes basic requirements and supplies basic facilities for the latter; whereas further generalization is required to generate on-demand maps according to specific user needs. This research focuses on the quality evaluation of the first generalization mode. The research objective also implies that the quality issues in thematic mapping and generalization of categorical data (i.e. area partitions such as land-use, vegetation, and forest data), as well as the evaluation of generalized data against user requirements from application domains, is out of the scope.

The topics covered in this research are driven by the research needs identified in previous work and also by the requirement for a meaningful evaluation as explained in Section 1.2.2. Previous efforts and emphasizes have primarily been put on the formulation and evaluation of readability constraints and limited preservation constraints on individual objects. Preservation constraints for groups of objects and feature classes were less discussed. On the other hand, a meaningful evaluation requires the use of preservation constraints for groups and feature classes (as can be found in the answer to question 2 in Section 8.1.2). In this context this research focuses especially on the preservation constraints for groups and feature classes, as well as topics (such as data matching) that come along as a result of evaluating such constraints. Nevertheless, it is not to say that the readability constraints and preservation constraints for individual objects are irrelevant. In fact, this research has proposed an improved automated evaluation framework

where all types of constraints are taken into account. It is just that some of the topic are not at the core of the discussion, as they are extensively studied in previous work on generalization evaluation.

This chapter summarizes the whole thesis. Section 8.1 concludes the work that has been done in this research and answers the research questions according to the research findings. Section 8.2 elaborates the contributions made in this research to related fields. Further improvements and recommendations for future work are pointed out in Section 8.3.

8.1 Conclusions

The conclusions start from a general conclusion (Section 8.1.1), followed by answers to research questions defined in Section 1.3 (Section 8.1.2), and end with reflections on specific topics (Section 8.1.3).

8.1.1 *Main conclusion*

Formalization of map specifications, data enrichment and data matching techniques form a promising framework and provide necessary instruments for automated evaluation of generalization output. As a starting point to the formalization, this research has expressed map specifications in cartographic constraints, which are suitable for specifying quality criteria for quality evaluation. Then constraints have been formalized in a predicate logic representation together with a spatial data model where spatial entities and relationships are formalized. This formalism helps to unambiguously define cartographic constraints for automated evaluation. In addition, a management issue for constraints has been highlighted in this research. That is, a set of evenly balanced constraints helps to enable a meaningful evaluation, e.g., to reduce over-optimistic evaluation (Section 3.4). The proposed framework has been demonstrated by automatically evaluating constraints on keeping building alignments and their properties (group level constraint), and on keeping spatial distributions of classes of objects in particular buildings and road networks (feature class level constraint). To do this, building patterns and alignments (implicit pattern entities) and their characteristics (implicit properties) have been automatically enriched for the group level constraints. For the feature class level constraint, indicators of spatial distributions and their similarities have been enriched automatically. For both types of constraints, (semi-)automated matching methods have been proposed and successfully applied to link initial entities (e.g. buildings and alignments) to their corresponding ones in target data. The study of the above constraints is motivated by the requirement for a

meaningful evaluation. Hence a more comprehensive evaluation based on the above preservation constraints together with individual level (readability) constraints (e.g. size, shape, position, etc.) in the future should lead to a meaningful evaluation of building class. Limitations have been discussed in the proposed solutions to framework components (Chapter 7) and is summarized in the answer to research questions (Section 8.1.2).

8.1.2 *Answers to research questions*

Research questions (Section 1.3) to reach the general objective are divided into two groups. The first group concerns the selection and formalization of map specifications for automated evaluation of generalization output (Questions 1 through 4). The second group concerns the components that are key to the automated evaluation process (Questions 5 through 8). Answers to these questions are provided below based on the research findings.

1. *What are the map specifications that define the quality of general-purpose map generalization and how should different map specifications be evaluated?*

   The answer to the first question is based on the background provided in Chapter 1 and a review of previous efforts in defining map specifications for generalization and quality evaluation (Section 2.3.2). To summarize, map specifications for general-purpose map generalization are derived from conventional map requirements for generalization which are usually maintained in the form of written map specifications at national mapping agencies (NMAs), as general-purpose topographic maps are the main product at NMAs. In this research, cartographic constraints are used to express map specifications for the evaluation. In general, two types of constraints, i.e., readability constraints which aim to improve to readability of the map and preservation constraints which aim to keep important characteristics of the map content, are used to evaluate the quality of cartographic generalization (Section 3.2.1). These two general constraints are also in correspondence with fundamental objectives of map generalization as described in generalization textbooks such as SSC [184]. Further, each of the two types of constraints can be defined on individual objects, groups, and feature classes (Section 3.2.2). In addition to the above classifications of cartographic constraints, which are derived basically from the outcome of the EuroSDR project on the state-of-the-art of automated generalization (Sections 2.3.2 and 2.4), Section 3.2.3 further distinguishes between property-/relation-level and entity-level constraints. Such a division is motivated by the strategies to evaluate the different types of constraints.

The ways in which different types of constraints are evaluated have been discussed in Section 3.2. In short, readability constraints are evaluated independent of source (initial) data/maps, while preservation constraints should be evaluated with respect to initial data. The conditions to be satisfied (strictly for hard constraints or as much as possible for soft constraints) are defined in each type of constraints. Besides, property-/relation-level preservation constraints only consider those spatial entities that have been kept in generalized (target) data, so that the evaluation of such constraints starts from entities in target data and refers back to their corresponding initial entities to see whether the changes are acceptable regarding the specifications. Examples include 'target shape should be similar to initial shape'. Entity-level preservation constraints, on the other hand, require that all the entities in a data set that are prescribed in a constraint should be kept (should not be deleted) during generalization. For example, 'alignments' in the constraint 'alignments should be kept'. Hence the evaluation of such constraints starts from entities in initial data and checks whether they are still represented in target data (existence checking). These different ways of evaluation should give insights into the design of evaluation procedures for each type of cartographic constraints.

2. *What are the problems in the selection of map specifications for automated evaluation and what are the principles in selecting the specifications for a meaningful evaluation?*

   The main problems in the selection of map specifications (constraints) have been described in Section 1.2 and evidence can also be found in Chapter 2 through 3. To conclude, inappropriate selection of map specifications (constraints) may be open to misinterpretations of evaluation results or lead to contradictions. In Section 3.4.1, I have carried out a hypothetical experiment in which a set of generalization outputs is evaluated (hypothetically) against a set of constraints. It has demonstrated that the use of only readability constraints led to over-optimistic evaluation conclusions, e.g., all objects were deleted from the initial data but the evaluation result report no constraint violation. It has demonstrated further that, by adding preservation constraints at the group and feature class level the evaluation bias or over-optimism can be reduced. Note that, although in the experiment the constraint violations were not really measured from the data but were estimated by visual assessments, it does not prevent readers to come to the same conclusion. Besides, it is not a good practice to use the same set of constraints to evaluate the generalization output as the constraints used to control the generalization process, as this may also lead to over-

optimistic evaluation, especially when the set of constraints is not evenly balanced. The over-optimistic evaluation was also mentioned in [88] and observed in [42].

A misleading or contradiction evaluation conclusion is meaningless. Hence a meaningful evaluation is characterized in this research as an evaluation that should avoid or minimize the biases and misinterpretations. According to Harrie and Weibel [88], a complete list of constraints can address the over-optimistic evaluation. However, a set of evenly balanced constraints is more realistic to collect. The guidelines for selecting map specifications for a meaningful evaluation have been proposed in Section 3.4.2, which is based on the literature review, the outcome from the EuroSDR project on automated generalization, as well as the hypothetical experiment described in Section 3.4.1. In essence, misleading evaluation results can by caused by the ways in which different types of constraints are evaluated (see the answer to question 1). Therefore, guidelines for a set of evenly balanced constraints actually balance between different types of constraints such as readability, property-/relation-level, and entity-level preservation constraints, and between different levels of granularity such as individuals, groups, and feature classes (Section 3.2). For detailed guidelines readers are referred to Section 3.4.2.

3. *How to formalize map specifications so that quality criteria for automated evaluation can be unambiguously expressed? In particular, how to formalize complex concepts and relationships prescribed in the specifications? Which formal concepts are useful in the formalization process?*

   Attempts to formalizing map specifications (constraints in this thesis) have been made in Chapters 3 and 4. To begin with, Section 3.2.4 has analyzed the syntactic structure of constraints from a linguistic point of view. Entity class, scope modifier, constrained property, and condition to be respected are identified as building blocks of the structure of constraints. Various types of constraints can be expressed using some combination pattern of the building blocks (syntactic terms). For a detailed explanation and examples, readers are referred to Section 3.2.4.

   Further, Chapter 4 has presented a formal theory, where a simplified entity-relation model for spatial data is sketched out. The entity-relation model essentially formalizes the above mentioned syntactic building blocks. First, the notion of spatial entity are formally given by a 2-tuple (Def. 4.1) which has a granularity attribute distinguishing between three levels, namely the micro-, meso-, and macro-level

entity. Def. 4.1 actually describes an abstract entity, and its corresponding concrete entities of the three levels of granularity are 'Object' (Def. 4.2), 'Group' (Def. 4.3) and 'Feature class' (Section 4.1.1). The three levels of entities can be characterized by their distinct feature class types and geometry types. These steps formalize the syntactic term — entity class — in Table 3.1. Second, ontological (semantic) and spatial relationships between entities and properties of entities are formalized as mathematic functions, where qualitative relationships such as topological relationships are in the form of functions with boolean output while quantitative relationships and properties (e.g. distance and size) are specified in functions with continuous output (see Def. 4.7). Consequently, semantic scope modifier in Table 3.1 is formalized as class (or type) predicate given in Def. 4.6; conditional scope modifier is formalized as conditional expressions; contextual scope modifier is formalized as contextual relationships defined in Def. 4.8. Likewise, property-/relation-level constrained property in Table 3.1 is formally expressed by mathematic functions. Third, corresponding relationships between entities of different scales are given in Def. 4.9, which helps to formalize constraints that have to preserve initial entities, characteristics, and relationships. Besides, mathematic (conditional and logical) expressions are used to formalize other parts of statements (such as conditions to be respect by constraints as described in Table 3.1). Finally, cartographic constraints are formalized in predicate logic based on an assembly of the previously mentioned formal constructs. Examples and the expressive power of the formal theory have been shown in Sections 4.3 and 4.4.

It is worth noting that there are also (loosely defined) spatial concepts in constraints which need to be well defined. In this research, I have mainly focused on the complex concepts in the form of group entities such as networks, clusters, and alignments. These concepts are pertinent to the generalization from large to mid scale for building and road feature classes (focused scale range and feature classes in this thesis). Section 4.1.4 has defined the concepts of network (Def. 4.10) and alignment (Def. 4.12) on top of the definition of group entity (Def. 4.3). In the proposed formal theory, a group of connected rive segments, or an alignment of buildings, is a meso-level entity and a class of objects is a macro-level entity. Complex entities have their own characteristics such as connectivity of networks, regularity of alignments, and spatial distribution of feature classes, which are modeled as properties of the entities. Relationships and properties of complex entities are also formalized as the above-mentioned functions. Such a treatment

enables to automatically evaluate the change of complex concepts at scale transitions (see e.g. the evaluation of building alignments in Section 7.5 and spatial distribution in Section 7.6).

To conclude, the presented formal theory (Chapter 4) is able to precisely define all the cartographic constraints that can be combined from the syntactic terms described in Section 3.2.4. The formal theory adopts concepts from set theory, predicate logic, and mathematic functions and expressions to formalize aspects of a multi-scale data model, including entities and relationships, and cartographic constraints. The formal concepts used in this research are pure mathematic so that the theory is implementation independent. It is nonetheless possible to use implementation specific formal languages such as the Object Constraint Language (OCL) and UML to realize the formal theory. However, OCL still lack of supporting tools, that is, the mechanism for evaluating OCL constraints is not powerful enough to be able to evaluate the quality of generalized maps defined by constraints. Besides, OCL constraints do not allow for the flexibility required by the generalization evaluation and assessments of fuzzy constraints.

4. *To what extent does knowledge obtained from experts or interactively (manually) generalized data help to formalize map specifications?*

   As a step before the formalization of map specifications, cartographic experts are usually involved in the definition of the specifications. They are also involved in specifying the parameters and thresholds for the constraints for their situations. However, there are ways that the experts can contribute to the formalization process. Since written map specifications are usually defined in an ambiguous manner and is, in most cases, only human readable, a knowledge acquisition process is always necessary for the formalization. Several examples in the implementation of the framework components in Chapter 7 has demonstrated the utility of expert knowledge and interactively generalized maps.

   Questionnaires and surveys are useful to extract in-depth knowledge on modifications of constrained properties that are acceptable for different scales. For example, Appendix B presents an expert survey designed to extract expert knowledge concerning the generalization of building patterns (Part II). In this research, a cartographer was invited to complete the survey, which yielded valuable information that helps the formalization of related constraints. For example, knowledge as to for which scales the regularity of building patterns should be preserved, improved, or reduced can be obtained. The survey can as

well answer whether building patterns should be transformed into built-up areas or be eliminated at all, for which scales and why. Such information is useful to define constraints on building patterns for different scales.

Analysis of interactively (manually) generalized data contributes also to the formalization of cartographic constraints. A statistical analysis of the interactively generalized (target) dataset alone sheds light on whether the threshold and tolerance in a readability constraint is appropriate, as is shown in [42, 192]. A regression analysis of linked initial and target objects (generalized interactively) gives insights into the specifications of acceptable modifications to their properties over scale transitions. The regression analysis is useful for formalizing preservation constraints, and therefore, it requires corresponding relationships be maintained or established by data matching. This technique was adopted in related work such as [171] and [42]. In this respect, Section 7.5.6 has demonstrated an analysis of interactively generalized data in formalizing constraints on preserving building alignments. This analysis has suggested that there is a need for improved formalization of constraints on building alignments. That is, another constraint that enhances building alignments at scale transitions should be considered in addition to the constraint that preserve the alignments.

5. *What are the requirements (or components) for an automated evaluation framework to be meaningful?*

   As in the answer to question 2, a complete or evenly balanced set of map specifications (constraints) can be used to reduce the over-optimistic evaluation and hence enables a meaningful evaluation. This statement, however, captures only part of the story. A meaningful evaluation system requires not only a component to manage and balance map specifications (Section 3.4) that defines the quality of generalization output, but also components to carry out the evaluation process (i.e. to formalize, measure, characterize, and evaluate the specifications). The latter requirement implies that an meaningful evaluation framework should also be equipped with basic components (e.g. evaluation procedures) and functional facilities (e.g. measuring algorithms) to be able to quantify and qualify generalized data or maps against formalized map specifications. Although there are three passages of evaluation (Section 3.1) and which passage to follow depends on the map specifications selected for a given task, a meaningful evaluation framework should provide an infrastructure (components and facilities) that is able to address the requirements

imposed by all the three passages. These requirements are described in the proposed conceptual framework for automated evaluation in Chapter 3, and in short, the framework requires:

- Formal map specifications. Formal specifications can be reviewed as an interface between intended users (e.g. cartographers) and an evaluation system. Ideally, a constraint management unit is required for users to express map requirements (or quality criteria) in formal specifications (constraints), to balance between the constraints and tune their relative importance, and to set constraint parameters (thresholds and tolerances). In other words, formal map specifications are at the core of an automated evaluation framework. Also depending on which constraints are evaluated, the following functional components may or may not be active (see the discussion in Section 3.1).
- A component to enrich existing topographic data. The quality evaluation in the generalization domain examines not only object-based information such geometries and attributes stored in a database but also information such as implicit characteristics and semantics of objects, relationships, and patterns. Many of the latter type of information can only be inferred from the representation. So an automated evaluation framework should be able to enrich existing data(bases) with such information before any meaningful evaluation can be performed. Actually, data enrichment is not only a requirement for automated evaluation of generalized data, it is also a prerequisite for automated map generalization [e.g. 49].
- A component to maintain or build corresponding relationships. To be meaningful, an automated evaluation framework should also be capable of evaluating preservation constraints. This in turn requires the corresponding relationships between spatial entities be explicitly available to the system. Ideally, automated evaluation should better be integrated with and carried out in an MRDB environment where links between corresponding objects at different scales are available. Reliable links can be created in a generalization process. For currently available datasets, a more practical way of creating the links is via a data matching process, which however rebuilds the corresponding relationships with degrees of uncertainty. The degree of uncertainty of corresponding relationships has been taken into account by the proposed formal theory (see Def. 4.9 in Section 4.1.3) and shown in the eval-

uation of the preservation constraints on building alignments (see Sections 6.4 and 7.5).

– A component to evaluate enriched topographic data. This is required to evaluate whether generalization output is satisfactory with respect to the constraints. Before this step, information needed for evaluating a constraint should be readily available either it is enriched by the above-mentioned components or it is encoded in existing data(bases). This research uses an approach to evaluate individual constraints that describe different quality aspects of generalized data instead of trying to develop an overall quality criteria. This seems to be the only way to automated the evaluation process since an overall quality criteria can be very difficult (if not impossible) to formalize. Individual assessments (e.g. constraint violations – CVs) can be aggregated at several levels. First, assessments of different constraints can be aggregated for a single entity. For example, by summarizing CVs of constraints on size, shape, orientation, and position defined for individual buildings, one can conclude if a generalized building object is generally acceptable or not. Second, assessments of a single constraint defined for individual entities (either an object or a group) can be aggregated to a single assessment (value) for a feature class to indicate the quality of the feature class concerning this particular constraint (as demonstrated in Sections 6.4 and 7.5). Then, different constraints defined for individuals can be firstly aggregated for the individuals and then summarized for the feature class to indicate the general quality of the generalized feature class. Finally, all the constraints defined for a dataset/map can be aggregated to indicate the quality of the dataset/map.

  The aggregation of different constraints assumes a linear relationship between constraints. Visualization of individual assessments (as demonstrated in Fig. 3.2 in Section 3.4.1) and visualization of evaluation on maps (as demonstrated in Section 6.4.2) can be used instead in case the assumption of the aggregation is problematic.

6. *Which types of information are required by automated evaluation but are not encoded in topographic data to be evaluated and how can the information be added to current data and evaluated?*

   The information that is not explicitly modeled and hence should be enriched has been discussed and demonstrated in Chapters 5 and

6. To summarize, several types of implicit information should be enriched:

(a) Implicit geometries which are always associated with implicit (higher-level) semantics. Implicit geometries are usually parts of or aggregations of existing geometries that can be identified by certain methods. For example, a peninsula is part of land in some coastal area; a residential area is an aggregation of houses in a neighborhood.

(b) Implicit semantics of existing objects or higher-level concepts such as isolated buildings or terraced houses.

(c) Meso-level entities (groups) that are geographically significant but are not explicitly modeled in the data model. Examples of meaningful groups are networks, patterns, and alignments.

(d) Implicit properties (characteristics) of, and relationships between, existing objects as well as between the first three types of implicit entities. Examples are orientations and local densities (implicit property) of buildings, parallel relationships (implicit relationship) between objects, homogeneities of building alignments (meso-level entity), and shapes of peninsulas (implicit geometry).

(e) Corresponding relationships between entities of different map scales.

Usually the above information can be added to current data either from other sources manually or it can be inferred automatically from existing data via data enrichment by making use of the implicit properties and relationships of exiting data objects. As a proof-of-concept, the following types of information regarding building and road features have been added (enriched) to topographic data in an automatic or semi-automatic way. The enriched information can be used for automated evaluation of the features at three levels of analysis (Chapter 5). First, properties of, and relationships between, individual objects such as distance, shape, orientation, and proximity have been designed. The properties and relationships and their changes at scale transitions are defined in map specifications and to be evaluated. Some of the measures are based on auxiliary structures such as Delaunay triangulation described in Appendix A. Second, significant meso-level entities, namely building clusters and alignments, have been identified as group entities with their properties (e.g. regularity, orientation, curvature descriptor, etc.) measured (Sections 5.3 and 5.4). This helps

to automatically evaluate constraints about alignments (Section 6.4). Third, local densities and spatial distributions of a set (feature class) of objects are enriched to the dataset to be evaluated. Density balance and spatial distribution are important characteristics to preserve for feature classes. Finally, to enable automated evaluation of preservation constraints, a generic as well as *ad-hoc* approach is developed to create links between corresponding spatial entities (Chapter 6). The *ad-hoc* approach is designed to automatically match building patterns and alignments, which is a prerequisite for automated evaluation of preservation of building alignments (Section 6.4).

7. *How can the key concepts and components in the proposed evaluation framework be validated and to what extent does knowledge obtained from experts or interactively generalized data help?*

   It is important for an evaluation approach to be validated or calibrated. This research has suggested a inner feedback loop carried out interactively by cartographic experts to calibrate an evaluation approach by tuning map specifications and parameters (e.g. thresholds and importance values) until the evaluation approach is satisfactory (as demonstrated by Fig. 3.4 in Section 3.5). Ideally, any evaluation system should be calibrated before application.

   Due to the time constraint, this research did not implement a whole evaluation system but focused on the specific aspects of a meaningful evaluation framework that have less been discussed in previous evaluation methods. Hence I only validated the key concepts and components to investigate their feasibility in future evaluation. The basic ideas of validating the implemented concepts and components were to ask cartographic experts and to test the implementations against interactively generalized data. In short, the building pattern recognition approach was validated by expert knowledge with ranges of parameters identified. For data matching approaches, the matching accuracy was validated against the matching performed by cartographic experts. For the evaluation of the constraints on building alignments (three constraints) and on spatial distribution, interactively generalized data and source data are used. Specifically, for each source data, good examples and bad (or nearly good) examples were prepared. The validation assumption is that if the evaluation results confirm the quality of the examples that are known beforehand, i.e., give higher scores (lower CVs) to good generalization examples and lower scores (higher CVs) to bad examples, the evaluation approach is feasible and effective. The validation issue has been demonstrated throughout Chapter 7.

8. *What are the capabilities and limitations of the proposed framework for automated evaluation?*

   This research has demonstrated the key concepts and components developed for automated evaluation of generalization output, where building feature class (road networks have been evaluated in terms of density and spatial distribution) has been used as a proof-of-concept. The evaluation has focused on group and feature class level preservation constraints which is required to balance the commonly used readability constraints for individuals. At the group level, three constraints on preserving alignments and their properties are evaluated; whereas at the feature class level a constraint on keeping spatial distribution of building class and network features are evaluated. Without the group and feature class level constraints, an evaluation may give misleading conclusions (as demonstrated in Section 3.4).

   By applying the framework to the focused thematic classes, Chapters 5, 6 and 7 have demonstrated the capabilities of the framework. To conclude, this research shows that it is feasible to evaluate the quality of generalized data/maps in a (semi-)automatic way, given that the requirements for a meaningful automated evaluation framework (as described in the answers to question 5 and 6) are fulfilled. It shows further (by evaluating constraints on building alignments) that the proposed framework has several merits:

   (a) it is able to rank (identify optimal or obviously poor) generalization outputs objectively according to the specifications;

   (b) it can be used for detecting and editing badly generalized situations;

   (c) it is able to inform intended users (e.g. cartographers) in case poor situations are detected or the system is not confident of its assessments, which in turn helps the users to identify and correct potential cartographic errors.

   In particular, this research has demonstrated that it is possible to evaluate geospatial patterns (building patterns and alignments in this research) in generalized data automatically and objectively based on a three-step evaluation (data enrichment, data matching, and evaluation). Object-oriented modeling has shown its potential in modeling and evaluating patterns and their behaviors over scale transitions (generalization of patterns).

   Due to the time constraint however, this research did not evaluate the quality of a whole map (i.e. a comprehensive set of constraints

defined for a full spectrum of feature classes). Given this deficiency, the requirements for a meaningful evaluation and the three-step evaluation approach (data enrichment, data matching, and evaluation) are still applicable for evaluating a whole map. On the other hand, it is still problematic to realize a fully automated system to evaluate the quality of a whole map (a fully automated evaluation is possible only if constraints are formalized and all types of required information is explicitly available and reliable). As shown in this research, although automated data enrichment and matching techniques have been adopted to address the issues that hamper the automation of the evaluation, these approaches have their own limitations that up to date fail to enable a fully automated evaluation process. The open issues mainly include the degree of automation and reliability of the approaches, which have been discussed in Chapter 7 for each of the approaches presented in this research. For example, automated recognition of geospatial patterns such as building alignments is not perfectly satisfactory, recognition quality and cognitive aspects of the recognition should be further investigated. The generic data matching approach (based on pattern classification) is in essence a supervised classification approach, which means that the evaluation based on such a matching technique is semi-automatic because the extra human effort is required to generate training samples. Moreover, the matching accuracy of the supervised classification approach is currently not satisfactory for a reliable evaluation of preservation constraints. Further development is possible to improve the matching accuracy (see Section 7.4.4). For the matching of building alignments, while 100% matching accuracy was reached by the *ad-hoc* matching algorithm (Section 6.3.3), the partial matching of the patterns reduces the reliability of the evaluation results. Currently this issue has been addressed by introducing a confidence indicator to describe the reliability of the evaluation results, but improvement can be done to increase the reliability of the evaluation results in the future (Section 8.3.2).

8.1.3 *Reflections on specific topics*

Confidence of automated evaluation decisions

This research has also identified a problem that an evaluation decision is not necessarily of 100% certainty. Hence to what degree an evaluation decision is reliable (certain) is a general issue that cannot be avoided. For example, an assessment may state that a building is well generalized, which in fact may not be the case. The problem can be caused from several sources.

First, it could be the case that a property or relationship constrained by a constraint cannot be measured in complete certainty. An example of this is the orientation of buildings. Since building orientations are not all well-defined, the measured orientations always contain some degrees of uncertainty [59]. Second, enriched geometries or recognized patterns may in nature be ambiguous (see the ambiguities involved in the recognition of building patterns Section 7.3.6). The third source of uncertainty comes from the data matching process. Corresponding relationships created via data matching are in essence uncertain [46]. In this research, uncertainties in data matching are in two forms. One is the result of the matching based on probabilistic judgement of criteria (Section 6.2), and the other uncertainty is caused by partial matching between alignments (Section 6.3.3).

Some of the uncertainties can be reduced by improving the techniques used but most of them cannot be eliminated. We therefore suggest that the evaluation system should be able to inform intended users of the uncertain information, if the information is available and quantifiable. The users then get a better awareness of the reliability of the assessments and can act on that information (e.g. to check if there is potential errors or not). In this research, the uncertainty information resulted from the data matching using e.g. Naive Bayes classifiers and the confidence indicator of partial matching can be propagated to described the confidence of the evaluation (see Sections 6.4.2 and 7.5 for the use of confidence indicator in describing the reliability of evaluation results).

8.2 Contributions

Although this research has addressed the problems in automated evaluation of generalization output, the work described in this thesis also contributes to the fields of automated generalization, multiple representation databases (MRDBs), as well as more general subfields in the geo-information domain such as spatial data integration and spatial data mining and knowledge discovery. The contributions are shortly described as follows.

- This research has identified the research needs from previous work on automated evaluation of generalization output and strengthen the idea for a meaningful evaluation. A meaningful evaluation requires to balance between readability constraints and preservations constraints and between constraints at individual, group, and feature class level. Previous researches primarily focused on readability constraints and some preservation constraints for individuals, hence may give rise to misinterpretations of evaluation results (see Sections 3.2.3 and

3.4.1). This research has demonstrated the possibility to automatically evaluate group and feature class level preservation constraints, where focused themes (buildings and networks) have been used as a proof-of-concept (Chapters 5 and 6).

- This research has proposed a more comprehensive evaluation framework in which data enrichment and data matching components are emphasized, i.e., the two components should be explicitly included as active units of the framework in addition to the final evaluation step (see the three-step evaluation in Chapter 3 and particularly Section 3.5). The three-step evaluation approach provides instruments for evaluating different types of map specifications required for a meaningful evaluation, where readability, preservation, entity-level preservation, property-/relation-level constraints, and constraints on individuals, groups and feature classes should be balanced (Section 3.2).

- This research has provided a formal theory for precisely defining map specifications (cartographic constraints) for the automated evaluation (Chapter 4). In this theory, aspects of an entity-relationship model are formalized, where spatial entities of different levels of granularity, intra-scale relationships (including contextual relationships), and inter-scale (corresponding) relationships are defined in a formal framework. This formal entity-relationship model is also valuable to model spatial concepts for the geospatial domain in general. With this model, cartographic constraints can be defined in an unambiguous and machine-readable way. The formalization of cartographic constraints not only contributes to automated generalization and evaluation, but also is highly relevant to the research on spatial integrity constraints for database applications [55, 56].

- In particular, this research has provided solutions to the quality evaluation of building and network features. The provided solutions follow the three-step evaluation approach and hence demonstrate the utility of the approach. The presented solutions contribute not only to automated evaluation of generalized data, but also to other related fields. The approaches described have some degree of flexibility and can be extended for other feature classes or used for other application domains.

  - Building pattern recognition in topographic data. This research has proposed an approach to automatically recognize building patterns from topographic data making use of implicit properties of and relationships between building instances. To begin with,

a typology of building patterns in clusters has been proposed based on studying maps. The recognition approach integrates aspects of computational geometry, graph-theoretic concepts, and theories of visual perception. The novelty lies in that it advocates a multi-algorithm paradigm, where a mechanism is proposed to combine results from different algorithms to improve the recognition completeness and quality. In addition, this research has developed a method to validate the recognition quality by investigating the precision and recall of automatically recognized patterns with respect to those recognized by experts. As the adopted perception theories are generic to geometric forms, the recognition approach can be applied to the detection of patterns in other geographic features, such as groups of ponds, lakes, and archipelago, as long as related recognition rules are modified according to their own characteristics.

The recognized patterns can be applied in many ways. For instance, the quality measure (i.e. homogeneity value) we derived for building patterns can be used to investigate the generalization of the patterns. These include enhancement of homogeneity, derivation of pattern hierarchies as well as aggregation, typification and simplification of patterns. Besides, the quality measure can be used for automated evaluation of whether and to what extent building patterns are preserved at scale transitions (see the following point). Furthermore, as structures and patterns can be used to infer underlying processes that generated the patterns (Section 2.1.1), building patterns and alignments can be used to indicate the functional aspect of the use of the land, e.g., whether it is a residential area, a holiday resort, an industrial complex, or a military base (which is demonstrated in [75] and the findings of the expert survey for building patterns). A foreseeable future of this topic could be that computers can automatically indicate the possible uses of lands inferred from automatically recognized structures and patterns based on a domain knowledge base. Such information may facilitate different cadastral and urban planning activities, and can support advanced spatial analysis and information retrieval, semantic enrichment and interoperability of data, etc.

– Data matching of spatial entities. The two basic problems in the matching of data instances (vs. matching of schema) are how to choose and design appropriate similarity criteria and how to effectively combine different criteria for decision making. This

research has proposed to fit the matching of data instances into a pattern classification framework to provide solution for the second issue, which is the major contribution of the data matching research in this thesis. The novelty lies in that the normalization and weighting process is not needed because model structures can be learnt from training samples. This research has shown that the tested classifiers (CART, C4.5, Naive Bayes and SVM) outperformed the weighted average approach to combining matching criteria. In the latter, the weighting of criteria is a trial-and-error process. With only four basic similarity criteria (i.e. position, size, shape and orientation), all the tested classifiers reached an average accuracy of approximately 80% and higher (Section 7.4.2). In addition, the matching based on probabilistic learning model (e.g. Naive Bayes) can be further improved by incorporating generalization knowledge (Section 7.4.3).

Additionally, this research has presented an *ad-hoc* algorithm to match building alignments. The matching algorithm explicitly deals with so-called *partial matching* and identifies corresponding parts for each matched pair of alignments. To indicate the degree of partial matching, a *confidence indicator* is proposed, which is valuable for both the matching and the subsequent evaluation process.

To summarize, two (semi-)automatic solutions have been provided to match spatial entities at micro- and meso-level respectively (semi-automatic for the generic matching approach and automatic for the *ad-hoc* matching algorithm). In addition to their utility in automated evaluation, the solutions also contribute to the field of MRDBs.

– Evaluation of building alignments in generalized data. This research has provided a solution to objectively and automatically evaluate a type of geospatial patterns (i.e. building alignments) at scale transitions. With explicit modeling (recognition) of alignment entities and matching of corresponding alignments of different map scales, the behavior (characteristics) of alignments are compared and analyzed with respect to the related map specifications. This evaluation method is also able to deal with many-to-many corresponding relationships between alignments. This research has argued that the reliability issue of assessments comes from many sources and are unavoidable, and therefore has suggested a need to explicitly address this issue (Section 8.1.3). The confidence indicator resulted from the

matching process are then used to indicate the reliability of the evaluation of alignments. This information helps intended users to identify potential cartographic errors that may be missed by the evaluation. Finally, the evaluation solution is also applicable to evaluate alignments of other features such as ponds, lakes, and archipelago.

– Similarity measure of spatial distributions. The constraint on keeping spatial distribution is a very important one for group and feature class level entities. In automated generalization, spatial distribution was implicitly handled during generalization operations such as displacement [17] and typification [41, 172, 173]. Although approaches were discussed to describe spatial distributions in terms of global patterns (i.e. random, regular, and clustered), they are not adequate for comparing the similarity between distributions (see a detailed discussed in Sections 5.5 and 5.6). This made the automated evaluation of related constraints hardly possible. This research has proposed two measures to explicitly compare and analyze the similarity between spatial distributions of groups of objects. The measures are capable of describing the similarity between spatial distributions of point, linear (e.g. networks), areal (building polygons) features. The effectiveness of the measures has been proven in Section 7.6 with limitations of each discussed in Section 7.6.3. In addition to the application in automated evaluation of generalization output, the similarity measures can be used for other purposes such as spatial analysis and environmental assessment.

8.3 Recommendations for future work

8.3.1 *General recommendations for automated evaluation*

Future automated evaluation of generalization output should take into account other feature classes and constraints for a meaningful evaluation for a whole map. This requires interrelationship between constraints and feature classes and the aggregation of assessments of different constraints be further investigated. Besides, the relationship between geospatial patterns and scales at which patterns become salient should be further studied. This is because future quality evaluation should focus more on higher-level phenomena, their characteristics and relationships, as already discussed in Sections 1.1.3 and 2.1. Among others, (transportation and river) networks are important structures at smaller scales. Currently, techniques were proposed to recognize kinds of network patterns and to generalize them [e.g.

90, 91, 92, 93, 196, 197, 198]. The quality evaluation of such group level entities still needs further investigation, which requires to formalize the characteristics and patterns of the features and their acceptable behavior at scale transitions.

8.3.2 *Possible improvements for current approaches*

The proposed conceptual framework for automated evaluation consists of different components. Approaches described in the components have limitations (see the answer to question 8) and can be improved in many ways.

Formal languages

As discussed in Section 4.4.4, implementation specific formal language such as the Object Constraint Languages (OCL) together with UML models have potentials to implement the formal theory and the formal constraints. However, OCL are lacking supporting tools for automated reasoning, which is essential for automated evaluation of OCL constraints. Currently, OCL constraints have to be translated into SQL queries to identify data instances that violate the constraints [194]. This is appropriate for hard constraints but not for soft constraints. In the future, other specification languages with more powerful reasoning capabilities can be sought as alternatives to implement the presented formal theory and constraint evaluation.

Data enrichment and matching

- **Recognition of building alignments:** As can be seen from Fig. 5.14, MST may disconnect desired proximity relationship. Hence at times the tracing procedure has to resort to the edges of the proximity graph. In extreme cases, the tracing procedure starting from MST may fail to recognize building patterns (e.g. the stream of buildings next to the road on the right side of Fig. 7.3(a)). An improvement could be obtained by starting the tracing procedure on the proximity graph instead of MST. This has at least three advantages. First, the tracing procedure is able to detect those extreme case alignments as more options are available to it. Second, the search-for-connection and search-for-extension described in Section 5.4.2 are no longer needed. Third, the iteration explained in Section 5.4.4 is not necessary as all alignments can be detected at the first round of tracing. This last point can avoid the problem caused by removing those shared buildings. The potential downside of this improvement is that it would

increase the searching space and hence the computational time. Besides, this improved approach may not be deterministic, because there are no fixed vertices for the tracing procedure to start from, and the tracing has to be adapted for two directional searching. In current development, the tracing starts from vertices of degree 1 and 3 and searches only in one direction. Therefore, further research is required to confirm the above merits in practice.

- **Similarity measures of spatial distribution:** Two methods have been introduced to measure the similarity between spatial distributions. They are density-based (Sections 5.5.1 and 6.5) and distance-based (Section 5.6) measures. The limitations of the two approaches have been given in Section 7.6.3.

  The density-based similarity measure considers only the densities of objects that were kept in the target dataset and hence densities of those removed objects are not evaluated. Such a measure is said to be based only on incomplete information. In the future, density distribution of feature classes can be converted to 2-dimensional density maps (images), the similarity measure could be based on the comparison between initial and target density images. This way, complete information concerning spatial distribution is made use of by the density-based measure.

  The distance-based measures, on the other hand, treat areal features such as building polygons as point-like features. This is an oversimplified model. Future development should take into account the size and shape and even semantics of the objects in the distribution, which should influence the perception of the spatial distribution. In addition, being a member of a significant group (e.g. cluster or alignment) also increases the importance of an object in determining the spatial distribution. A same idea concerning the weighting of objects in a distribution was presented in [41] for building typification with an attempt to preserve spatial distributions. Further research should be carried out to improve the similarity measures of spatial distributions that takes different weighting factors into account.

- **Data matching:** The generic data matching approach based on pattern classification has the potential to match spatial entities of different LODs. More matching criteria such as contextual similarity [103, 169] and semantic similarity [153] can be incorporated into learning algorithms to further improve the matching so that a satisfactory accuracy can be obtained.

Evaluation

- **Evaluation of building alignments:** It is possible to reduce the uncertainty caused by the partial matching issue by designing an alternative evaluation workflow. Take the evaluation on the spatial distribution of composing buildings for example, to improve its reliability one can (i) recognize alignments only in target data, (ii) identify initial buildings that correspond to the buildings that compose the target alignment, forming a group (not necessary an alignment) that corresponds to the target alignment; and (iii) measure homogeneity values of the identified group and the target alignment and compare the values according to the constraint. If a reliable linkage between individual buildings is maintained, a reliable evaluation could be obtained. However, reliable links are only possible when they are created during generalization [46]. On the other hand, the constraints should be formalized in a different way to imply the way how the evaluate is carried out (including how to deal with the partial matching and n-to-m relationship). Besides, other requirements such as on enhancing the regularity of alignments (Equ. 7.1) should be formalized and further considered by the evaluation.

- **Aggregation of individual assessments:** The evaluation approach decomposes quality of generalization output into different aspects (e.g. geometry, topology, pattern, distribution and statistics), after which an aggregation process is anticipated. Currently, the aggregation of individual assessments into a single value assumes that the constraints used can be linearly ordered, which could be problematic. Alternatively, the interrelationship among constraints can be modeled in a semantic network as suggested by AGENT [1]. Technically, it is possible to model different quality criteria in Bayesian networks, and the aggregation can be done using the built-in probabilistic reasoning of the networks [see e.g. 99]. This idea provides a possibility to model cartographic constraints and their interrelationship in a nonlinear model.

Auxiliary geometric structures

This chapter gives preliminary information concerning the auxiliary data structures that are the fundation of many of the approaches proposed in Chapters 5 and 6. Two basic structures are Delaunay triangulations (Section A.1) and a Voronoi-like diagram (Section A.2).

Own publications

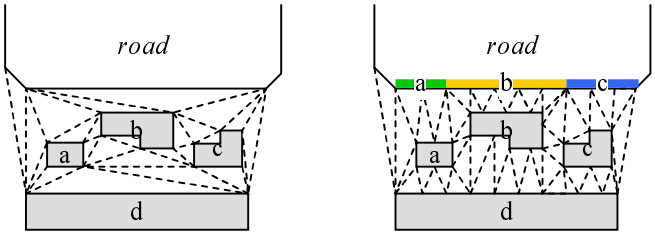
This chapter is partly based on the following publications:

- T. Ai and X. Zhang. The aggregation of urban building clusters based on the skeleton partitioning of gap space. In S. I. Fabrikant and M. Wachowicz, editors, *The European Information Society: Leading the Way with Geo-information (AGILE07)*, Lecture Notes in Geoinformation and Cartography, pages 153–170. Springer, Aalborg, Denmark, 2007.
- X. Zhang, T. Ai, and J. E. Stoter. A voronoi-like model of spatial autocorrelation for characterizing spatial patterns in vector data. In *Proceedings of the 2009 Sixth International Symposium on Voronoi Diagrams (ISVD09)*, pages 118–126, Washington, DC, USA, 2009. IEEE Computer Society.

A.1 Delaunay triangulations

Delaunay triangulation (DT) has been widely used in the geospatial domain for modeling and analyzing purposes. It also proves to be a powerful tool in analyzing shape of spatial objects and proximity relationships between objects in automated map generalization [2, 3, 8, 15, 100, 206] and was used to implement generalization operations on either individual objects or object groups [7, 9, 17, 41, 172, 173].

Section A.1.1 discusses the choices that can be made to calculate DTs and



(a) CDT (b) Refined CDT

Figure A.1 Differences between constraint Delaunay triangulation (CDT) (a) and refined CDT (b)

justifies our choice of implementation. In Section A.1.2, I describe several measures and operations that can be calculated with the DT structure.

A.1.1 *Options for triangulations*

There are several choices to make in order to construct DTs for different application purposes, each with subtle differences in their properties and modeling capabilities. First, it is arguable whether to use centroid points of areal features or points of their outlines to construct DT. Evidently, using centroids is more efficient in computation but is less effective in modeling accuracy. Using outlines to calculate DT better describes the proximity relationship between linear and areal objects. A second consideration is how to deal with the relationship between triangle edges and edges of objects. Usually, constrained Delaunay triangulation (CDT) is used to avoid the intersection between triangles and object outlines, where edges in outlines are added as constrained lines.

However, CDT is not always able to correctly capture relationships between spatial objects. For example in Fig. A.1(a), the proximity relationship between buildings b and d is not modeled by the CDT. This usually occurs when very long edges are constrained in CDT, leading to very narrow triangles that block the proximity relationship. A mathematically elegant solution to this problem is the so-called *conforming (Gabriel) triangulation* which is a special sort of DT [158]. Every CDT can be refined into conforming (Gabriel) triangulation by adding extra vertices (called *Steiner vertices*). Steiner vertices are added to constrained lines and the positions of which are determined by criteria to control the shape and size of output triangles [158]. Hence calculate conforming triangulation is even time-consuming (more than CDT and DT).

In this research, an alternative approach is used to address the problem of narrow triangles in CDT. Extra vertices are added to constrained lines (e.g.

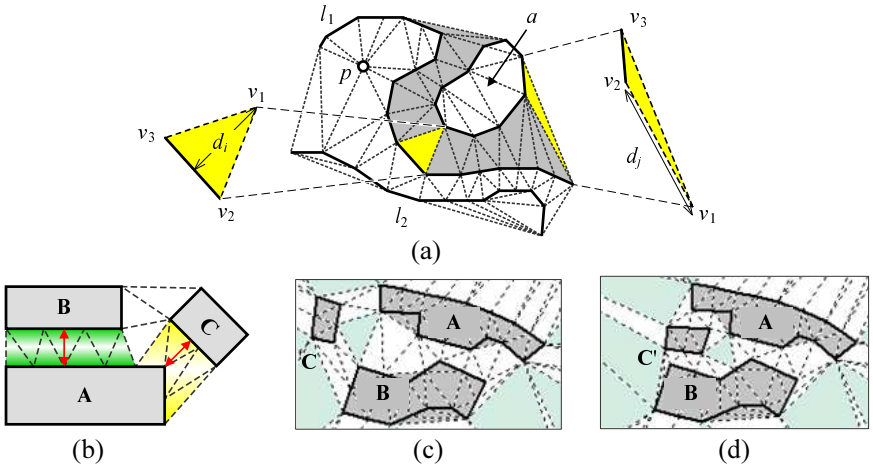


Figure A.2 Refined CDT applied to point, linear, and areal features (a), and illustrations of weighted average distance (b)–(d)

building outlines and road lines) to refine the CDT. The positions of the inserted vertices are pre-specified depending on the characteristic of the dataset. That is, the interval of insertion is based on the minimum distance between data points. This approach may take up more (run-time) memory but is more efficient and easier to implement, and provides an approximation to the conforming Delaunay triangulation. After the refinement, proximity relationship is better captured. In Fig. A.1(b), the proximity relationship between b and d are now correctly captured (i.e. they are connected by triangles). In addition, the refined CDT is able to identify the parts of the road to which the buildings align (see the colored bold segments in Fig. A.1(b)). This is especially useful when detailed information between proximate buildings and roads are needed. Note also that this refined CDT is also CDT since the resulting triangles do not intersect edges of the objects.

A.1.2 *Measures and operations*

The refined CDT can be calculated for a mixture of different types of geometries (Fig. A.2(a)). In this section, I describe several measures and operations that can be performed over the DT structure. To simplify the presentation, we use notations from point-set topology [62]. For example, we use t_i, t_i°, and ∂t_i to denote a triangle, its interior, and its boundary, respectively.

Proximity

The immediate proximity relationship between two spatial objects can be defined based on the triangles in the DT:

$$ImdProx(o_i, o_j) \equiv (\exists t_i)\,((\partial t_i \cap o_i = \neg\emptyset) \wedge (\partial t_i \cap o_i = \neg\emptyset))$$

For example, $ImdProx(p, l_1)$, $ImdProx(p, l_2)$, $ImdProx(l_1, l_2)$, and $ImdProx(a, l_1)$ hold in Fig. A.2(a). As such, the triangles involved in the relationship between two immediately proximate objects can be defined as a set of triangles:

$$T_{\langle o_i, o_j \rangle} = \{t_i | (\partial t_i \cap o_i = \neg\emptyset) \wedge (\partial t_i \cap o_i = \neg\emptyset)\}$$

For example, the triangles involved in the relationship between a and l_1, $T_{\langle a, l_1 \rangle}$, are highlighted in gray in Fig. A.2(a).

Distances

Several distance measures can be computed from the refined CDT. The measures rely entirely on the involved triangles between two proximate objects, $T_{\langle o_i, o_j \rangle}$. Each triangle $t_i \in T_{\langle o_i, o_j \rangle}$ has one and only one vertex (v_1) which coincides with one object, and has the reminding two vertices (v_2, v_3) which coincide with the other object. A distance d_i can be calculated for t_i from v_1 to the segment defined by v_2 and v_3. Note that if the perpendicular incidence from v_1 does not intersect the segment $\overline{v_2v_3}$, d_i equals to the minimum length of $\overline{v_1v_2}$ and $\overline{v_1v_3}$. The calculation of d_i is illustrated by the yellow triangles in Fig. A.2(a).

This way, the nearest distance between two objects (Section 5.2.1) can be implemented using this refined CDT structure as: $NearestDis(A, B) = \min(\{d_i\})$, where d_i is measured from $t_i \in T_{\langle A, B \rangle}$. In theory, this implementation is computationally more efficient than the brute force implementation. The time complexity of the DT implementation is $\mathcal{O}(k)$, where k is the number of incident triangles between two objects. A brute force implementation (as was used in [154]) takes $\mathcal{O}(n \times m)$ time complexity, where n and m are the numbers of the points of both objects respectively. Since k is generally much smaller than $n \times m$ that can be expected between two objects (Fig. A.2(a)) and especially between an building and a road, the DT implementation greatly accelerates the computation of nearest distance measure.

Then weighted average distance between the objects can be implemented based on $T_{\langle o_i, o_j \rangle}$. It is defined as follows:

$$WADis(o_i, o_j) = \frac{\sum_{t_i \in T_{\langle o_i, o_j \rangle}} d_i \times w_i}{\sum_{t_i \in T_{\langle o_i, o_j \rangle}} w_i} \tag{A.1}$$

where d_i is the above mentioned distance measured for each t_i in $T_{\langle o_i, o_j \rangle}$; the weighting factor w_i is the length of the edge ($\overline{v_2v_3}$ in Fig. A.2(a)) of t_i, that is, the edge that coincides with one of the two objects. This distance measure can be used to address the issue of nearest distance that arises from cognitive aspects (see Section 5.2.1). For example, two pair of objects with the same nearest distance but are perceived to be different. In Fig. A.2(b) the nearest distance between A and B equals to that between A and C. According to Equ. A.1, one can calculate that $WADis(A, B) < WADis(A, C)$ ($T_{\langle A, B \rangle}$ and $T_{\langle A, C \rangle}$ are shaded in green and yellow, respectively, in Fig. A.2(b)), meaning that the weighted average distance better reflects the cognitive difference between perceived distances. In addition, the weighted average distance is a contextual measure. This can be demonstrated by Fig. A.2(c) and (d), where objects A and B stay unchanged while C was displaced to C′ in Fig. A.2(d). The calculated $WADis(A, B)$ will definitely change simply because the triangles involved in the relationship between A and B, $T_{\langle A, B \rangle}$, changes. To be specific, weighted average distance between A and B becomes smaller after C moved to C′. Of course, the proposed weighted average measure of distance should be further tested against human subjects to see to what extent it is cognitively appropriate.

Furthermore, it is possible to measure the degree of parallelism between two (linear and/or areal) objects. The parallelism can be based on the variance of the distances d_i measured for $t_i \in T_{\langle o_i, o_j \rangle}$. For example, if two lines are completely parallel to each other, it is obvious that STD($\{d_i\}$) equals to zero; with the increase in the variance, the two lines become less parallel.

A.2 A Voronoi-like diagram based on skeleton

This section present a structure that mimic the Voronoi diagram. The proposed structure is called Voronoi-like diagram which is the approximation of Voronoi diagram for linear and areal features. This structure can be used to calculate the impact areas of features and hence is useful in calculating local density described in Section 5.5.1.

The Voronoi-like diagram is calculated based on the notion of (DT-based) skeleton. The DT skeletons are used to partition the gap space between spatial objects. The partition of gap space for points is degenerated into the calculation of Voronoi diagram, and thus is not described here. In the fol-

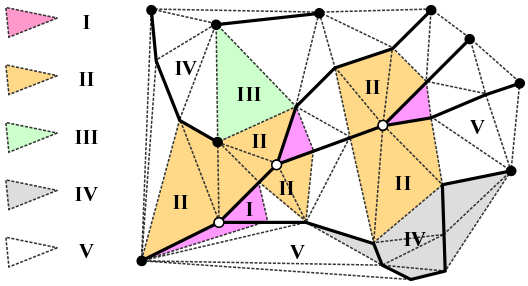


Figure A.3 Classification of triangles for skeletonization (○ are nodes and ● are end points of linear features)

lowing we mainly focus on the discussion of partitioning network features (i.e. curves), since as shown later this section that other cases (e.g. polygons) can be generalized from the procedure.

A.2.1 *Skeletonizing gap space*

This section describes the procedure to skeletonize the gap space between objects to calculate Voronoi-like diagram. I firstly give a formal definition of gap space, then describe a classification of the triangles in refined CDT for the skeletonization. At last, the procedure to skeleton the gap space is presented.

Gap space is the space between spatial objects. In triangulation terms, gap space can be represented by a set of triangles $T_{gap} = \{t_i \,|\, (\forall o_i)(t_i^\circ \cap o_i = \emptyset)\}$.

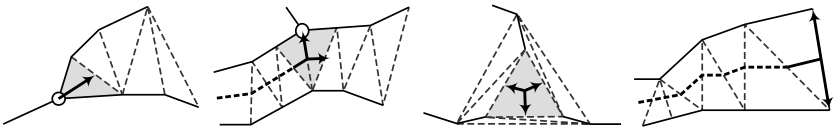
Classification of triangles

Five types of triangles that are important for the skeletonization procedure are identified and illustrated in Figure A.3. The triangle types are defined as follows:

T-I This is a t_i that satisfies: $(\exists n_i)(\partial t_i \cap n_i = \neg\emptyset)$ and $|\{l_i \,|\, \dim(\partial t_i \cap l_i) = 1\}| \geq 2$, meaning that at least one *node*, n_i, coincides with a vertex of t_i, and edges from at least two curves, l_i, coincide with the edges of t_i.

T-II This is a t_i that satisfies: $(\exists n_i)(\partial t_i \cap n_i = \neg\emptyset)$ and there is at most one edge of t_i which coincides with the line segment of one of the feature connected to n_i.

T-III This is a t_i that relates at least three objects. More precisely, the three vertices of t_i link to objects that are different from each other, and $(\forall n_i)(\partial t_i \cap n_i = \emptyset)$.



(a) Type I triangle (b) Type II triangle (c) Type III triangle (d) Boundary case

Figure A.4 Four skeletonization cases

T-IV This is a t_i that has all its three vertices coincided with a some curve.

T-V This is a t_i that is not of the above-mentioned types (see white triangles in Figure A.3).

Note that, for refined CDT that is calculated for areal features, only type III, IV, and V triangles are possible. This is due to the absence of *node* in the topology of disjoint polygons.

Skeletonizing procedure

At the implementation level, a data structure recording basic information of skeleton is as follows. The useful information are the `LeftObject` and `RightObjecct` that record the IDs of linear or areal objects on the left and right side of the skeleton.

```
Skeleton
{p1, p2, ..., pn}
LeftObject
RightObject
WalkthroughTriangles
```

Partitioning is based on the skeletonization of all types of triangles except the T-IV triangles, because this type of triangles is always inside polygons or the bends of curves, and is therefore regarded to be the private space of these geometries. Four skeletonization cases are described as follows (see also Figure A.4).

First of all, new skeleton points are usually determined at the middle point of triangle edges (notated as *MP*). The skeletonization of a T-I triangle is

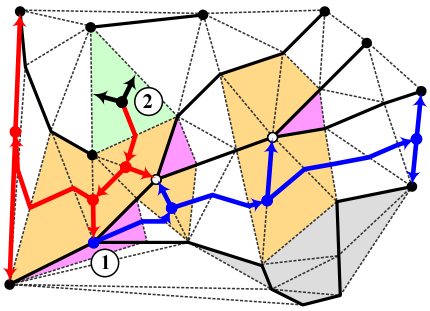


Figure A.5 Examples of partitioning scheme for a network feature

shown in Fig. A.4(a). The skeleton starts from a *node* coinciding with the triangle, to the opposite edge on the triangle. A T-II triangle is a place where a skeleton has to be split into two branches, as the skeleton is entering a place occupied by two objects (Figure A.4(b)). Likewise, a skeleton starting from the *barycenter* of a T-III triangle always have three branches (Figure A.4(c)), since the triangle is occupied by three objects. When a skeleton is approaching to the boundary of the refined CDT, as is depicted in Figure A.4(d), it must be divided into two branches pointing toward opposite directions. The skeletons of T-V triangles are formed by consecutively connecting the *MPs* as walking through the T-V triangles. For the skeleton being created, its left and right hand side objects should be recorded in the data structure described above.

The next step is to put all the skeletonization cases into a partitioning algorithm described as follows and demonstrated in Fig. A.5.

1. The partitioning starts from a T-I triangle and ends with a T-III, another T-I triangle, or the boundary of refined CDT (No. 1 in Fig. A.5);

2. Alternately, it starts from a T-III triangle and ends with a T-I, another T-III triangle, or the boundary of refined CDT (No. 2 in Fig. A.5);

3. During step 1, and 2, the skeletonization cases described in Fig. A.4(b) must be deployed in case of entering T-II triangles.

4. The algorithm repeats steps 1–3 until all the T-I and -III triangles have been visited.

Note that for the partition of polygons, step 1 and 3 are ignored, as there is no triangle of T-I and -II in such a case.

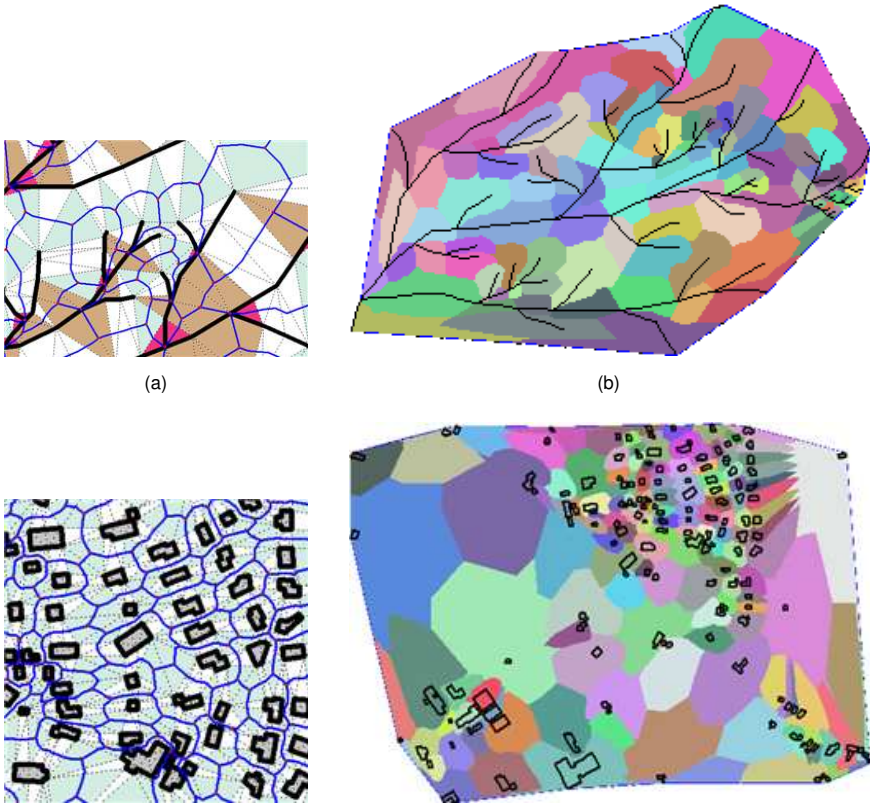


Figure A.6 Calculated skeletons (a) and (c), and Voronoi-like diagram (b) and (d)

A.2.2 *Organizing diagram from skeletons*

Using the information collected during the skeletonization process, e.g. the left and right objects associated with each skeleton, the impact area (unit cell) of an object is formed by connecting adjacent skeletons around the object.

The resulting partition is the so-called Voronoi-like diagram. The Voronoi-like diagram for various geometry types is demonstrated in Fig. A.6). An observation of the partitions indicates that there is an intrinsic relationship between the distribution of the partition units and the pattern of the original geometries. Hence they can be used as an indirectly indicator to characterize the spatial patterns for these geometries.

B

Expert survey of building patterns for generalization

Background and purpose

Settlement structures can be found at different levels of observation which are important features on topographic maps. At a global level, we can define different contexts of built environments, e.g., urban, rural, mountain areas, etc. Furthermore, they can be divided into residential, town center, industrial/commercial areas, etc. This survey focuses primarily on local level structures/patterns of building groups (i.e. arrangements of building groups), as they are fundamental features to operate when generalizing building features while keeping settlement structures at other levels. Example building patterns are collinear, curvilinear, align-along-road alignments, grid-like, unstructured clusters (see a detailed explanation in Section 5.3.1).

This survey aims at extracting expert knowledge concerning building group structures/patterns that are important for map generalization processes. It consists of two parts. The first part aims to extract expert knowledge about building patterns and to compare the building patterns visually recognized and those recognized by the algorithms developed in Section 5.4. The second part aims to get insight into how cartographers generalize building patterns on topographic maps. At the end of each part, the participants also have the choice to mark if they think the answers they provided are subjective (applicable only for their own NMAs) or neutral (applicable for any NMA).

Please indicate your NMA or organization[1]: _____

[1] This information is used for analyzing purposes only and will not be open to the public.

PART I. BUILDING PATTERNS ON TOPOGRAPHIC MAPS

1. Which of the above-mentioned pattern types do you think are important and/or if you know other types not mentioned here (multiple answers possible)?

   □ Collinear alignment

   □ Curvilinear alignment

   □ Align-along-road alignment

   □ Grid-like pattern

   □ Other types: _____

2. Why do you think these building patterns are important (optional)?

   _____

   _____

   _____

   There are three map clips on the following pages, please mark the building structures which you consider as importance with a (color) pen.

3. Please mark building patterns on map (a): part of a topographic dataset of the Netherlands (1:50k)

Please answer:

3.1. Which type(s) of pattern do you observe on map (a)?

3.2. Which type(s) of pattern do you observe most on map (a)?

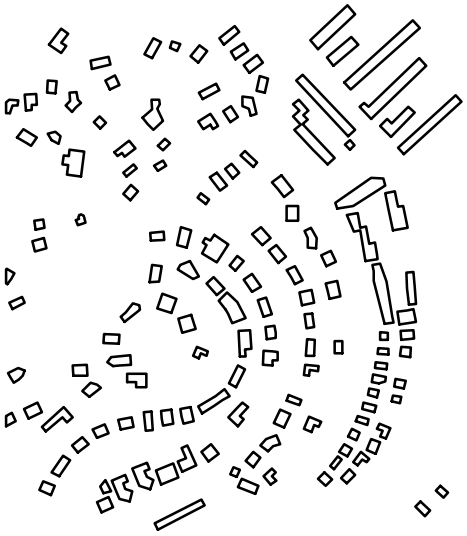
4. Please mark building patterns on map (b): part of a topographic dataset of the Netherlands (1:50k)

Please answer:

4.1. Which type(s) of pattern do you observe on map (b)?

4.2. Which type(s) of pattern do you observe most on map (b)?

5. Please mark building patterns on map (c): part of a topographic dataset of ICC, Catalonia, Spain (1:25k)



Please answer:

5.1. Which type(s) of pattern do you observe on map (c)?

5.2. Which type(s) of pattern do you observe most on map (c)?

6. Do you find it is sometimes difficult to distinguish between different types of building patterns (e.g. between collinear, curvilinear and align-along-road alignments)?

□ Yes (continue with question 6.1)

□ No

6.1. What are the possible reasons for this ambiguity? (multiple answers possible)

□ The boundary between different pattern types is ambiguous (e.g. a collinear alignment could also be an align-along-road alignment).

□ One pattern (of one type) can be part of another pattern (of another type).

□ Other reasons: _____

Please indicate to which questions your answers are subjective or neutral:

	Q 1	Q 2	Q 3	Q 3.1	Q 3.2	Q 4	Q 4.1	Q 4.2
Subjective	□	□	□	□	□	□	□	□
Neutral	□	□	□	□	□	□	□	□

	Q 5	Q 5.1	Q 5.2	Q 6	Q 6.1
Subjective	□	□	□	□	□
Neutral	□	□	□	□	□

PART II. GENERALIZATION OF BUILDING PATTERNS[2]

1. Please specify the scale range in which the building patterns can be observed in your NMA or organization: _____

2. Does your organization have specifications which are related to the generalization of building structures?

   □ Yes

   □ No

3. When generalizing building maps, do you consider building structures explicitly?

   □ Yes (continue with question 3.1)

   □ No

   3.1. How do you think the building structures should be generalized in terms of regularity[3] and from 1:10k to each of the following target scale? (indicate where the scale is defined in your organization; multiple answers possible)

	1:25k	1:50k	1:100k	1:250k
Regularity improved	□	□	□	□
Regularity preserved	□	□	□	□
Regularity reduced	□	□	□	□

4. If you choose 'Yes' to Q 2 and 'No' to Q 3, how do you meet the specifications related to the generalization of building structures?

   _____

   _____

   _____

5. Do you generalize building patterns into built-up areas and for which target scales (e.g. via aggregation operation)?

[2] Focused scales: 1:10k, 1:25k, 1:50k, 1:100k and 1:250k

[3] Regularity improved means the building pattern becomes more regular; vice versa.

Yes				No
1:25k	1:50k	1:100k	1:250k	
☐	☐	☐	☐	☐

5.1. If yes, why do you generalize them into built-up areas?

6. Do you eliminate building patterns and for which target scales?

Yes				No
1:25k	1:50k	1:100k	1:250k	
☐	☐	☐	☐	☐

6.1. If yes, why do you eliminate building patterns?

Please indicate to which questions your answers are subjective or neutral:

	Q 1	Q 2	Q 3	Q 3.1	Q 4	Q 5	Q 5.1	Q 6	Q 6.1
Subjective	☐	☐	☐	☐	☐	☐	☐	☐	☐
Neutral	☐	☐	☐	☐	☐	☐	☐	☐	☐

Bibliography

[1] AGENT. Constraint analysis. Technical Report A2, Agent Consortium, 1998.

[2] AGENT. Selection of basic measures. Technical Report C1, Agent Consortium, 1999. URL `http://http://agent.ign.fr/deliverable/DC1.html`.

[3] T. Ai. The drainage network extraction from contour lines for contour line generalization. *ISPRS Journal of Photogrammetry and Remote Sensing*, 62(2):93–103, June 2007. ISSN 0924-2716.

[4] T. Ai and P. van Oosterom. Spatial relation abstract in map generalization process. In W. Aref, editor, *Proceedings of the 9th ACM International Symposium on Advances in Geographic Information Systems (ACM-GIS 2001)*, pages 21–29, Atlanta GA, 2001.

[5] T. Ai and P. van Oosterom. Gap-tree extensions based on skeletons. In D. Richardson and P. van Oosterom, editors, *Advances in Spatial Data Handling (SDH 2002)*, pages 501–513. Ottawa, 2002.

[6] T. Ai and P. van Oosterom. Displacement methods based on field analysis. In *Joint Workshop on Multi-Scale Representations of Spatial Data, IAPRS XXXIV(2)*, page 10 p., 2002.

[7] T. Ai and X. Zhang. The aggregation of urban building clusters based on the skeleton partitioning of gap space. In S. I. Fabrikant and M. Wachowicz, editors, *The European Information Society: Leading the Way with Geo-information (AGILE07)*, Lecture Notes in Geoinformation and Cartography, pages 153–170. Springer, Aalborg, Denmark, 2007.

[8] T. Ai, R. Guo, and Y. Liu. A binary tree representation of bend hierarchical structure based on gestalt principles. In *Proceedings of the 9th International Symposium on Spatial Data Handling*, pages 230–243, Beijing, 2000.

[9] T. Ai, Y. Liu, and J. Chen. The hierarchical watershed partitioning and data simplification of river network. In A. Riedl, W. Kainz, and

G. A. Elmes, editors, *Progress in Spatial Data Handling*, pages 617–632. Springer, 2006.

[10] T. Ai, Y. Shuai, and J. Li. The shape cognition and query supported by fourier transform. In A. Ruas, C. Gold, W. Cartwright, G. Gartner, L. Meng, and M. P. Peterson, editors, *Headway in Spatial Data Handling*, Lecture Notes in Geoinformation and Cartography, pages 39–54. Springer, 2008.

[11] C. Alexander. *The Phenomenon of Life: Nature of Order, Book 1: An Essay on the Art of Building and the Nature of the Universe*. The Center for Environmental Structure, Berkeley, California, 2002.

[12] H. Alt and M. Godau. Computing the fréchet distance between two polygonal curves. *International Journal of Computational Geometry and Applications*, 5(1-2):75–91, 1995.

[13] K.-H. Anders. A hierarchical graph-clustering approach to find groups of objects. In *The 5th Workshop on Progress in Automated Map Generalization*, Paris, 2003. URL `http://www.geo.unizh.ch/ICA/docs/paris2003/papers03.html`.

[14] K.-H. Anders. Grid typification. In A. Riedl, W. Kainz, and G. A. Elmes, editors, *Progress in Spatial Data Handling*, pages 633–642. Springer, Vienna, 2006.

[15] K.-H. Anders and M. Sester. Parameter-free cluster detection in spatial databases and its application to typification. In *proceedings of International Archives of Photogrammetry and Remote Sensing, Vol. 33(Part B4/1), ISPRS Congress*, pages 75–82, 2000.

[16] M. P. Armstrong. Knowledge classification and organization. In B. P. Buttenfield and R. B. McMaster, editors, *Map Generalization: Making Rules for Knowledge Representation*, pages 86–102. London: Longman, 1991.

[17] M. Bader, M. Barrault, and R. Weibel. Building displacement over a ductile truss. *International Journal of Geographical Information Science*, 19(8):915–936, 2005.

[18] C. Barber, R. Cromley, and R. Andrle. Evaluating alternative line simplification strategies for multiple representations of cartographic lines. *Cartography and Geographic Information Science*, 22(4):276–290, 1995.

[19] S. Bard. Quality assessment of cartographic generalisation. *Transactions in GIS*, 8:63–81, 2004.

[20] S. Bard. *Méthode d'évaluation de la qualité de données géographiques généralisées: Application aux données urbaines*. PhD thesis, Université de Paris, 2004.

[21] S. Bard and A. Ruas. Why and how evaluating generalised data? In *Developments in Spatial Data Handling (SDH' 04)*, pages 327–342. Springer, Leicester, 2004.

[22] M. Barrault, N. Regnauld, C. Duchêne, K. Haire, C. Baeijs, Y. Demazeau, P. Hardy, W. A. Mackaness, A. Ruas, and R. Weibel. Integrating multi-agent, object-oriented, and algorithmic techniques for improved automated map generalization. In *Proceedings of the 20th international cartographic conference (ICC 2001)*, pages 2110–2116, Beijing, China, 6-10 August 2001. (CD-ROM).

[23] M. K. Beard. Constraints on rule formation. In B. P. Buttenfield and R. B. McMaster, editors, *Map Generalization: Making Rules for Knowledge Representation*, pages 121–135. London: Longman, 1991.

[24] C. Beeri, Y. Doytsher, Y. Kanza, E. Safra, and Y. Sagiv. Finding corresponding objects when integrating several geo-spatial datasets. In *Proceedings of the 13th annual ACM international workshop on Geographic information systems*, pages 87–96, New York, NY, USA, 2005. ACM. ISBN 1-59593-146-5.

[25] L. Bejaoui, F. Pinet, M. Schneider, and Y. Bédard. OCL for formal modelling of topological constraints involving regions with broad boundaries. *GeoInformatica*, 14(3):353–378, 2010.

[26] T. Berners-Lee. An parenthetical discussion to the web architecture at 50,000 feet and the semantic web roadmap, September 1998. URL `http://www.w3.org/DesignIssues/RDFnot.html`. [Accessed: November 2011].

[27] E. Bernier and Y. Bédard. A data warehous strategy for on-demand multiscale mapping. In W. A. Mackaness, A. Ruas, and L. T. Sarjakoski, editors, *Generalisation of Geographic Information: Cartographic Modelling and Applications*, pages 177–198. Elsevier, 2007.

[28] R. Billen and E. Clementini. Semantics of collinearity among regions. In *On the Move to Meaningful Internet Systems 2005: OTM Workshops (SeBGIS 2005)*, volume 3762 of *Lecture Notes in Computer Science*, pages 1066–1076. Springer, 2005.

[29] T. Bittner and B. Smith. A theory of granular partitions. In M. Duckham, M. Goodchild, and M. Worboy, editors, *Foundations of Geographic Information Science*, pages 117–151. Taylor & Francis, London, 2003.

[30] T. Bittner, M. Donnelly, and B. Smith. Individuals, universals, collections: On the foundational relations of ontology. In A. Varzi and L. Vieu, editors, *Proceedings of the Third International Conference on Formal Ontology in Information Systems (FOIS04)*, pages 37–48. IOS Press, 2004.

[31] J. T. Bjørke. Framework for entropy-based map evaluation. *Cartography and Geographic Information Science*, 23(2):78–95, 1996.

[32] J. T. Bjørke. Map generalization: Information theoretic approach to feature elimination. In *Proceedings of the 18th International Cartographic Conference (ICC'97)*, volume 1, pages 480–486, Stockholm, 1997. Swedish Cartographic Society.

[33] J. T. Bjørke. Generalization of road networks for mobile map services: An information theoretic approach. In *Proceedings International Cartographic Conference (ICA)*, Durban, South Africa, 2003.

[34] M. Bobzien, D. Burghardt, I. Petzold, M. Neun, and R. Weibel. Multi-representation databases with explicitly modeled horizontal, vertical, and update relations. *Cartography and Geographic Information Science*, 35(1):3–16, January 2008.

[35] A. Boffet and S. Rocca Serra. Identification of spatial structures within urban block for town qualification. In *ICC*, volume 3, pages 1974–1983, Beijing, 2001.

[36] B. N. Boots and A. Getis. *Point pattern analysis*, volume 8 of *Scientific Geography series*. SAGA Publications, 1988. ISBN: 0-8039-2588-3.

[37] G. Borruso. Network density and the delimitation of urban areas. *Transactions in GIS*, 7(2):177–191, 2003.

[38] F. Brazile. *Semantic Infrastructure and Methods to Support Qualty Evaluation in Cartographic Generalization*. PhD thesis, Department of Geography, University of Zurich, 2000.

[39] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and regression trees*. Wadsworth & Brooks/Cole Advanced Books & Software, Monterey, CA, 1984.

[40] V. Bruce, R. P. Green, and A. Geogreson. *Visual Perception physiology, psychology, ecology*. Psychology Press, New York, 4 edition, 2003.

[41] D. Burghardt and A. Cecconi. Mesh simplification for building typification. *International Journal of Geographical Information Science*, 21(3): 283 – 298, 2007.

[42] D. Burghardt and S. Schmid. Constraint-based evaluation of automated and manual generalised topographic maps. In G. Gartner and F. Ortag, editors, *Cartography in Central and Eastern Europe*, Lecture Notes in Geoinformation and Cartography, pages 147–162. Springer-Verlag, 2010.

[43] D. Burghardt and S. Steiniger. Usage of principal component analysis in the process of automated generalisation. In *Proceedings of the XXII International Cartographic Conference*, 2005. (CD-ROM).

[44] D. Burghardt, S. Schmid, and J. Stoter. Investigations on cartographic

constraint formalization. In *10th ICA Workshop on Generalisation and Multiple Representation*, page 16, Moscow, August 2-3 2007.

[45] D. Burghardt, S. Schmid, C. Duchêne, J. Stoter, B. Baella, N. Regnauld, and G. Touya. Methodologies for the evaluation of generalised data derived with commercial available generalisation systems. In *The 11th Workshop of ICA Commission on Generalisation and Multiple Representation*, Montpellier, 20–21 June 2008.

[46] D. Burghardt, I. Petzold, and M. Bobzien. Relation modelling within multiple representation databases and generalisation services. *Cartographic Journal, The*, 47(3):238Ű249, 2010.

[47] B. P. Buttenfield. A rule for describing line feature geometry. In B. P. Buttenfield and R. B. McMaster, editors, *Map Generalization: Making Rules for Knowledge Representation*, pages 150–171. London: Longman, 1991.

[48] M. Carmona, T. Heath, T. Oc, and S. Tiesdell. *Public Places, Urban Spaces: The Dimensions of Urban Design*. Oxford: Architectural Press, 2003.

[49] A. Cecconi. *Integration of Cartographic Generalization and Multi-Scale Databases for Enhanced Web Mapping*. PhD thesis, Department of Geography, University of Zurich, Switzerland, 2003.

[50] O. Z. Chaudhry and W. A. Mackaness. Automatic identification of urban settlement boundaries for multiple representation databases. *Computers, Environment and Urban Systems*, 32(2):95–109, 2008.

[51] Y.-H. Chou. Critical issues in the evaluation of spatial autocorrelation. In *Spatial Information Theory A Theoretical Basis for GIS*, volume 716 of *LNCS*, pages 421–433. Springer Berlin / Heidelberg, 1993.

[52] Y.-H. Chou. Spatial pattern and spatial autocorrelation. In *Spatial Information Theory A Theoretical Basis for GIS*, volume 988 of *LNCS*, pages 365–376. Springer Berlin / Heidelberg, 1995. doi: 10.1007/3-540-60392-1_24.

[53] S. Christophe and A. Ruas. Detecting building alignments for generalisation purposes. In D. E. Richardson and P. van Oosterom, editors, *Advances in Spatial Data Handling (SDH 2002)*, pages 419–432. Springer Verlag, Ottawa, 2002.

[54] E. Clementini, P. Di Felice, and P. van Oosterom. A small set of formal topological relationships suitable for end-user interaction. In D. Abel and B. Chin Ooi, editors, *Advances in Spatial Databases*, volume 692 of *Lecture Notes in Computer Science*, pages 277–295. Springer Berlin / Heidelberg, 1993.

[55] S. Cockcroft. A taxonomy of spatial data integrity constraints. *GeoIn-*

formatica, 1(4):327–343, 1997.

[56] S. Cockcroft. The design and implementation of a repository for the management of spatial data integrity constraints. *GeoInformatica*, 8(1): 46–69, 2004.

[57] G. Dettori and E. Puppo. How generalization interacts with the topological and geometric structure of maps. In M. J. Kraak and M. Molenaar, editors, *Advances in GIS Research II (SDHŠ96)*, pages 559–570. Taylor & Francis, Delft, 1996.

[58] T. Devogele, J. Trevisan, and L. Raynal. Building a multi-scale database with scale transition relationships. In M. J. Kraak and M. Molenaar, editors, *Advances in GIS Research II (SDH'96)*, pages 337–351. Taylor & Francis, 1996.

[59] C. Duchêne, S. Bard, X. Barillot, A. Ruas, J. Trévisan, and F. Holzapfel. Quantitative and qualitative description of building orientation. In *5th Workshop on Progress in Automated Map Generalization*, Pairs, 2003. URL `http://www.geo.unizh.ch/publications/beatp/ICA99_Working_Group_Paper.pdf`.

[60] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2nd edition, 2001.

[61] A. Edwardes and W. A. Mackaness. Modelling knowledge for automated generalisation of categorical maps – a constraint based approach. In P. Atkinson and D. Martin, editors, *GIS and Geocomputation (Innovations in GIS 7)*, pages 161–173. Taylor & Francis, London, 1999.

[62] M. J. Egenhofer. Point-set topological spatial relations. *International Journal of Geographical Information Systems*, 5(2):161–174, 1991.

[63] M. J. Egenhofer. A model for detailed binary topological relationships. *Geomatica*, 47(3&4):261–273, 1993.

[64] M. J. Egenhofer and R. D. Franzosa. On the equivalence of topological relations. *International Journal of Geographical Information Science*, 9(2): 133–152, 1995.

[65] R. Ehrliholzer. Quality assessment in generalization: integrating quantitative and qualitative methods. In *Proceedings of the 17th International Cartographic Conference*, pages 2241–2250, Barcelona, 1995.

[66] H. B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, 1972.

[67] T. M. Field, A. Hayes, and R. F. Hess. Contour integration by the human visual system: Evidence for a local "association field". *Vision Research*, 33(2):173–193, January 1993.

[68] T. Foerster. *Web-based Architecture for On-Demand Maps - Integrating Meaningful Generalization Processing*. PhD thesis, International Insti-

tute for Geo-Information Science and Earth Observation, Enschede, the Netherlands, 2009.

[69] T. Foerster, J. M. Morales, and J. E. Stoter. A classification of generalisation operators formalised in OCL. In E. Pebezma and T. Bartoschek, editors, *Proceedings of GI-days 2008: interoperability and spatial processing in GI applications*, pages 141–156, Münster, Germany, 2008.

[70] T. Foerster, J. Stoter, and M.-J. Kraak. Challenges for automated generalisation at european mapping agencies: A qualitative and quantitative analysis. *Cartographic Journal, The*, 47(1):41–54, February 2010.

[71] A. Friis-Christensen, C. S. Jensen, J. P. Nytun, and D. Skogan. A conceptual schema language for the management of multiple representations of geographic entities. *Transactions in GIS*, 9(3):345–380, 2005.

[72] M. Galanda. Modelling constraints for polygon generalization. In *Proceedings 5th ICA Workshop on Progress in Automated Map Generalization*, Paris, 2003.

[73] M. Galanda and R. Weibel. An agent-based framework for polygonal subdivision generalization. In D. Richardson and P. van Oosterom, editors, *Advances in Spatial Data Handling (SDH 2002)*, pages 121–136. Springer-Verlag, Ottawa, 2002.

[74] C. Gangodagamage, X. Zhou, and H. Lin. Spatial autocorrelation. In S. Shekhar and H. Xiong, editors, *Encyclopedia of GIS*, pages 32–37. Springer, 2008.

[75] P. J. Gersmehl. *The Language of Maps*. Pathway in Geography Series. National Council for Geographic Education, 1991. ISBN: 0-9627379-3-3.

[76] C. M. Gold. The meaning of "neighbour". In A. U. Frank, I. Campari, and U. Formentini, editors, *Spatio-Temporal Reasoning*, volume 639 of *Lecture Notes in Computer Science*, pages 220–235. Springer, 1992.

[77] M. Goodchild and R. Jeansoulin. *Data quality in geographic information, from error to uncetainity*. Hermes, Paris, 1998.

[78] N. Guarino, S. Pribbenow, and L. Vieu. Modeling parts and wholes. *Data and Knowledge Engineering*, 20(3):257–258, 1996. (Editorial).

[79] S. Guptill and J. Marrison. *Elements of spatial data quality*. Elsevier, Oxford, 1995.

[80] V. Haarslev and R. Möller. SBox: A qualitative spatial reasoner - progress report -. In *11th International Workshop on Qualitative Reasoning*, pages 105–113, 1997.

[81] V. Haarslev, R. Möller, and C. Schröder. Combining spatial and terminological reasoning. In B. Nebel and L. Dreschler-Fischer, editors,

Advances in Artificial Intelligence, volume 861 of *Lecture Notes in Artificial Intelligence*, pages 142–153. Springer-Verlag, 1994.

[82] V. Haarslev, C. Lutz, and R. Möller. Foundations of spatioterminological reasoning with description logics. In A. Cohn, L. Schubert, and S. Shapiro, editors, *Principles of Knowledge Representation and Reasoning (KR'98)*, pages 112–123, 1998.

[83] M. Hampe, K.-H. Anders, and M. Sester. MRDB applications for data revision and real-time generalisation. In *Proceedings of the 21st International Cartographic Conference*, pages 192–202, Durban, South Africa, 2003.

[84] L. Harrie. *An Optimisation Approach to Cartographic Generalisation*. PhD thesis, Lund Institute of Technology, 2001.

[85] L. Harrie. Weight-setting and quality assessment in simultaneous graphic generalization. *Cartographic Journal, The*, 40(3):221–233, 2003.

[86] L. Harrie and T. Sarjakoski. Simultaneous graphic generalization of vector data sets. *GeoInformatica*, 6(3):233–261, 2002.

[87] L. Harrie and H. Stigmar. An evaluation of measures for quantifying map information. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(3):266–274, 2010.

[88] L. Harrie and R. Weibel. Modelling the overall process of generalisation. In W. Mackaness, A. Ruas, and L. Sarjakoski, editors, *Generalisation of geographic information: Cartographic modelling and applications*, pages 67–87. Elsevier, Amsterdam, 2007.

[89] L. E. Harrie. The constraint method for solving spatial conflicts in cartographic generalization. *Cartography and Geographic Information Science*, 26(1):55–69, 1999.

[90] F. Heinzle and K.-H. Anders. Characterising space via pattern recognition techniques: Identifying patterns in road networks. In W. A. Mackaness, A. Ruas, and L. T. Sarjakoski, editors, *Generalisation of Geographic Information: Cartographic Modelling and Applications*, pages 233–253. Elsevier, 2007.

[91] F. Heinzle, K.-H. Anders, and M. Sester. Graph based approach for recognition of patterns and implicit information in road networks. In *Proceedings of 22nd International Cartographic Conference (ICC2005)*, La Coruna, Spain, 2005.

[92] F. Heinzle, K.-H. Anders, and M. Sester. Pattern recognition in road networks on the example of circular road detection. In M. Raubal, H. J. Miller, A. U. Frank, and M. F. Goodchild, editors, *Geographic, Information Science*, volume 4197 of *Lecture Notes in Computer Science*, pages 153–167, 2006.

[93] F. Heinzle, K.-H. Anders, and M. Sester. Automatic detection of pattern in road networks - methods and evaluation. In *Proceedings of Joint Workshop Visualization and Exploration of Geospatial Data (IAPRS)*, volume XXXVI - 4/W45, 2007.

[94] P. Højhot. Solving space conflicts in map generalization: Using a finite element method. *Cartography and Geographic Information Science*, 27(1):65–74, Jan. 2000.

[95] J. Illian, A. Penttinen, H. Stoyan, and D. Stoyan. *Statistical Analysis and Modelling of Spatial Point Patterns*. Statistics in Practice. John Wiley & Sons, Ltd, 2008.

[96] M. Jansen and M. van Kreveld. Evaluating the consistency of cartographic generalization. In *The 8th International Symposium on Spatial Data Handling*, pages 668Ű–678, 1998.

[97] B. Jiang and C. Claramunt. A structural approach to the model generalization of an urban streets network. *GeoInformatica*, 8(2):157–171, 2004.

[98] E. João. *Causes and consequences of map generalisation*. Taylor & Francis, 1998.

[99] M. Jobst and F. Twaroch. An evaluation method for determining map-quality. In A. Riedl, W. Kainz, and G. A. Elmes, editors, *Progress in Spatial Data Handling*, pages 293–304. Springer, Vienna, 2006. ISBN 978-3-540-35589-2. doi: 10.1007/3-540-35589-8_19.

[100] C. Jones, G. Bundy, and J. Ware. Map generalization with a triangulated data structure. *Cartography and Geographic Information Systems*, 22(4):317–331, 1995.

[101] S. Kazemi, S. Lim, and L. Ge. Integration of cartographic knowledge with generalization algorithms. In *Proceedings of IGARSS'05*, volume 5, pages 3502–3505, 2005.

[102] T. Kilpeläinen. Maintenance of multiple representation databases for topographic data. *Cartographic Journal, The*, 37(2):101–107, 2000.

[103] J. Kim, K. Yu, J. Heo, and W. Lee. A new method for matching objects in two different geospatial datasets based on the geographic context. *Computers & Geosciences*, 36:1115–1122, 2010.

[104] M.-J. Kraak and F. Ormeling. *Cartography: Visualization of Spatial Data*. Longman, 1996.

[105] S. Lamy, A. Ruas, Y. Demazeau, M. Jackson, W. Mackaness, and R. Weibel. The application of agents in automated map generalisation. In *Proceedings of the 19th International Cartographic Conference*, volume 2, pages 1225–1234, Ottawa, 1999.

[106] D. Lee and G. Sallee. A method of measuring shape. *Geographical*

Review, 60(2):555–563, 1970.

[107] M. Leitner and B. P. Buttenfield. Acquisition of procedural cartographic knowledge by reverse engineering. *Cartography and Geographic Information Science*, 22:232–241, 1995.

[108] R. Lemmens. *Semantic interoperability of distributed geo-services*. PhD thesis, International Institute for Geo-Information Science and Earth Observation (ITC), Enschede, The Netherlands, 2006.

[109] C. Lévi-Strauss. *Structural Anthropology*. New York: Basic Books, 1963.

[110] Z. Li and P. Huang. Quantitative measures for spatial information of maps. *International Journal of Geographical Inforamtion Science*, 16(7): 699–709, 2002.

[111] Y. Liu. *Categorical Database Generalization in GIS*. PhD thesis, Wageningen University/ITC, the Netherlands, 2002.

[112] P. Lüscher, D. Burghardt, and R. Weibel. Ontology-driven enrichment of spatial databases. In *The 10th ICA Workshop on Generalisation and Multiple Representation*, page 13, Moscow, 2007.

[113] P. Lüscher, R. Weibel, and W. A. Mackaness. Where is the terraced house? on the use of ontologies for recognition of urban concepts in cartographic databases. In A. Ruas and C. Gold, editors, *Headway in Spatial Data Handling (SDH'08)*, Lecture Notes in Geoinformation and Cartography, pages 449–466. Springer, 2008.

[114] P. Lüscher, R. Weibel, and D. Burghardt. Integrating ontological modelling and bayesian inference for pattern classification in topographic vector data. *Computers, Environment and Urban Systems*, 33(5):363–374, 2009.

[115] K. Lynch. *The image of the city*. Publications of the Joint Center for Urban Studies. M.I.T. Press (Cambrigh, Mass), 1960.

[116] A. M. MacEachren and J. H. Ganter. A pattern identification approach to cartographic visualization. *Cartographica*, 27(2):64–81, 1990.

[117] W. A. Mackaness. Integration and evaluation of map generalisation. In B. P. Buttenfield and R. B. McMaster, editors, *Map Generalization: Making Rules for Knowledge Representation*, pages 217–226. London: Longman, 1991.

[118] W. A. Mackaness. Analysis of urban road networks to support cartographic generalization. *Cartography and Geographic Information Science*, 22:306–316, 1995.

[119] W. A. Mackaness and G. Edwards. The importance of modelling pattern and structure in automated map generalization. In *Joint Workshop on Multi-Scale Representations of Spatial Data*, page 11, Ottawa, Canada, 2002.

[120] W. A. Mackaness and A. Ruas. Evaluation in the map generalisation process. In W. A. Mackaness, A. Ruas, and L. T. Sarjakoski, editors, *Generalisation of Geographic Information: Cartographic Modelling and Applications*, Series of International Cartographic Association, pages 89–111. Elsevier Science, Amsterdam, 2007.

[121] W. A. Mackaness, A. Ruas, and L. T. Sarjakoski. *Generalisation of Geographic Information: Cartographic Modelling and Applications*. Elsevier Science, Amsterdam, 2007.

[122] D. Mark and M. Egenhofer. UCGIS emerging research theme: Ontological foundations for geographic information science. Technical report, UCGIS, 2000.

[123] D. M. Mark, A. Skupin, and B. Smith. Features, objects, and other things: Ontological distinctions in the geographic domain. In *in Spatial Information Theory: Foundations of Geographic Information Science*, LNCS, pages 488–502. Springer-Verlag, 2001.

[124] S. Marshall. *Streets & Patterns*. Spon Press, Taylor & Francis Group, New York, 2005.

[125] A. Mascret, T. Devogele, I. L. Berre, and A. Hénaff. Coastline matching process based on the discrete fréchet distance. In A. Riedl, W. Kainz, and G. A. Elmes, editors, *Proceedings of the 12th International Symposium on Spatial Data Handling*, pages 383–400. Springer, 2006.

[126] I. Matsuda. River morphology and channel processes. In J. C. Dooge, editor, *Fresh Surface Water*, pages 299–309. Encyclopedia of Life Support Systems (EOLSS), 2009.

[127] K. L. McGraw and K. Harbison-Briggs. *Knowledge acquisition, principles and guidelines*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[128] R. B. McMaster. *Mathematical measures for the evaluation of simplified lines on maps*. PhD thesis, Université du Kansas, Etats-Unis, 1983.

[129] R. B. McMaster. Automated line generalization. *Cartographica*, 2(24): 74–111, 1987.

[130] R. B. McMaster. Knowledge acquisition for cartographic generalization. In M. J. C., J. P. Lagrange, and R. Weibel, editors, *GIS and Generalization: Methodology and Practice*, pages 161–179. Taylor & Francis, 1995.

[131] R. B. McMaster and K. S. Shea. Cartographic generalisation in a digital environment: A framework for implementation in a geographic information system. In *GIS/LIS proceedings*, pages 240–249, San Antonio, TX, 1988.

[132] R. B. McMaster and K. S. Shea. *Generalization in Digital Cartography*. Resource Publications in Geography. Association of American

Geographers, 1992.

[133] M. Molenaar. *An Introduction into the Theory of Spatial Object Modelling for GIS*. Taylor & Francis, 1998.

[134] M. A. Mostafavi, G. Edwards, and R. Jeansoulin. An ontology-based method for quality assesment of spatial data bases. In *Proceedings of ISSDQ04*, pages 49–66, 2004.

[135] J. C. Müller. Generalization of spatial data bases. In D. J. Maguire, M. F. Goodchild, and D. W. Rhind, editors, *Geographical Information Systems: Principles and Applications*, pages 457–475. London: Longman, 1991.

[136] J. C. Müller. Building knowledge tank for rule based generalization. In *Proceedings of the 15th Conference of the International Cartographic Association*, 1991.

[137] J. C. Müller, R. Weibel, J. P. Lagrange, and F. Salgé. Generalization: state of the art and issues. In M. J. C., J. P. Lagrange, and R. Weibel, editors, *GIS and Generalization: Methodology and Practice*, pages 3–17. Taylor & Francis, 1995.

[138] S. Mustière. Cartographic generalization of roads in a local and adaptive approach: A knowledge acquisition problem. *International Journal of Geographical Inforamtion Science*, 19:937–955, 2005.

[139] S. Mustière and T. Devogele. Matching networks with different levels of detail. *Geoinformatica*, 12(4):435–453, 2008.

[140] S. Mustière and B. Moulin. What is spatial context in cartographic generalisation? In *Symposium on Geospatial Theory, Processing and Applications*, volume 34-B4, pages 274–278, Ottawa, 2002. ISPRS & SIS.

[141] J. Neumann. The topographical information content of a map / an attempt at a rehabilitation of information theory in cartography. *Cartographica*, 31(1):26–34, 1994.

[142] M. Neun. *Data Enrichment for Adaptive Map Generalization Using Web Services*. PhD thesis, Department of Geography, University of Zurich, 2007.

[143] M. Neun and S. Steiniger. Modelling cartographic relations for categorical maps. In *Proceedings of XXII International Cartographic Conference*, 2005.

[144] M. Neun, D. Burghardt, and R. Weibel. Automated processing for map generalization using web services. *GeoInformatica*, 13(4):425–452, 2009.

[145] B. G. Nickerson. Knowledge engineering for generalization. In B. P. Buttenfield and R. B. McMaster, editors, *Map Generalization: Making Rules for Knowledge Representation*, pages 40–55. London: Longman,

1991.

[146] M. S. Nixon and A. S. Aguado. *Feature Extraction and Image Processing*. Newnes, Oxford, 1st edition, 2002.

[147] J. Odland. *Spatial autocorrelation*, volume 9 of *Scientific Geography series*. SAGA Publications, 1988. ISBN: 0-8039-2651-0.

[148] B. Peter. Measures for the generalization of polygonal maps with categorical data. In *Fourth ICA Workshop on Progress in Automated Map Generalization*, Beijing, 2001.

[149] B. Peter and R. Weibel. Using vector and raster-based techniques in categorical map generalization. In *Third ICA Workshop on Progress in Automated Map Generalization*, 1999.

[150] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.

[151] C. Plazanet, N. M. Bigolin, and A. Ruas. Experiments with learning techniques for spatial model enrichment and line generalization. *GeoInformatica*, 2(4):315–333, 1998.

[152] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993. ISBN 1-55860-238-0.

[153] A.-M. O. Raimond and S. Mustière. Data matching – a matter of belief. In *Headway in Spatial Data Handling*, Lecture Notes in Geoinformation and Cartography, pages 501–519. Springer, 2008.

[154] N. Regnauld. Recognition of building clusters for generalization. In M. J. Kraak and M. Molenaar, editors, *Advances in GIS Research II (SDH'96)*, pages 4B.1–4B.14. Taylor & Francis, Delft, 1996.

[155] N. Regnauld. *Généralisation du bâti: Structure spatiale de type graphe et representation Cartographique*. PhD thesis, Université de Provence, Aix-Marseille, 1998.

[156] N. Regnauld. Contextual building typification in automated map generalization. *Algorithmica*, 30(2):312–333, 2001.

[157] T. Reichenbacher. Knowledge acquisition in map generalization using interactive systems and machine learning. In *Proceedings 17th International Cartographic Conference*, pages 2221–2230, Barcelona, 1995.

[158] L. Rineau. 2D conforming triangulations and meshes. In C. E. Board, editor, *CGAL User and Reference Manual*. 3.4 edition, 2008. URL `http://www.cgal.org/Manual/3.4/doc_html/cgal_manual/packages.html#Pkg:Mesh2`.

[159] M. E. Ritter. The physical environment: an introduction to physical geography, 2006. URL `http://www.uwsp.edu/geo/faculty/`

ritter/geog101/textbook/title_page.html.

[160] G. J. Robinson. A hierarchical top-down bottom-up approach to topographic map generalization. In J. Müller, J. Lagrange, and R. Weibel, editors, *GIS and Generalization: Methodology and Practice*, pages 235–245. Taylor & Francis, 1995.

[161] P. Rogerson. *Statistial Methods for Geography: A Student Guide*. Pine Forge Press, 2 edition, 2006. ISBN: 1412907950.

[162] A. Ruas. OO-Constraint modelling to automate urban generalisation process. In *Proceedings of the 8th International Symposium on Spatial Data Handling*. Springer Verlag, 1998.

[163] A. Ruas. The roles of meso objects for generalization. In *Proceedings of the 9th International Symposium on Spatial Data Handling*, pages 3B50–3B63, Beijing, China, 2000.

[164] A. Ruas. Automatic generalisation research: Learning process from interactive generalisation. Technical Report 39, OEEPE, 2001.

[165] A. Ruas and F. Holzapfel. Automatic characterisation of building alignments by means of expert knowledge. In *ICC*, pages 1604–1515, Durban, 2003.

[166] A. Ruas and C. Plazanet. Data and knowledge modeling for generalization. In M. J. C., J. P. Lagrange, and R. Weibel, editors, *GIS and Generalization: Methodology and Practice*, pages 73–90. Taylor & Francis, 1995.

[167] A. Ruas and C. Plazanet. Strategies for automated generalization. In M. J. Kraak and M. Molenaar, editors, *Advances in GIS Research II (SDH'96)*, pages 319–336. Taylor & Francis, 1996.

[168] Y. Sadahiro. Perception of spatial dispersion in point distributions. *Cartography and Geographic Information Science*, 27(1):51–64, 2000.

[169] A. Samal, S. Seth, and K. Cueto. A feature-based approach to conflation of geospatial sources. *International Journal of Geographical Information Science*, 18(5):459–489, 2004.

[170] L. T. Sarjakoski. Conceptual models of generalisation and multiple representation. In W. A. Mackaness, A. Ruas, and L. T. Sarjakoski, editors, *Generalisation of Geographic Information: Cartographic Modelling and Applications*, Series of International Cartographic Association, pages 11–35. Elsevier, Amsterdam, 2007.

[171] S. Schmid. Automated constraint-based evaluation of cartographic generalization solutions. Master's thesis, Department of Geography, University of Zurich, 2008.

[172] M. Sester. Optimization approaches for generalization and data abstraction. *International Journal of Geographical Inforamtion Science*, 19(8):

871–897, 2005.

[173] M. Sester and C. Brenner. Typification based on kohonen feature nets. In *Preccedings of the 1st International Conference on Geographic Information Science*, pages 21–22, Savannah, Georgia, USA, 2000.

[174] C. E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423 & 623–656, 1948.

[175] S. Shekhar, P. Zhang, Y. Huang, and R. R. Vatsavai. Trends in spatial data mining. In H. Kargupta, A. Joshi, K. Sivakumar, and Y. Yesha, editors, *Data Mining: Next Generation Challenges and Futhure Directions*, chapter 3, pages 357–380. AAAI Press, 2004.

[176] W. Shi, P. F. Fisher, and M. F. Goodchild, editors. *Spatial Data Quality*. Taylor & Francis, 2002.

[177] A. Shortridge. Practical limits of moran's autocorrelation index for raster class maps. *Computers, Environment and Urban Systems*, 31: 362–371, 2007.

[178] A. Skopelity and T. Lysandros. A methodology for the assessment of generalization quality. In *The Fourth Workshop on Progress in Automated Map Generalization*, Beijing, 2001.

[179] A. Skopelity and L. Tsoulos. Developing a model for quality assessment of linear features. In *Proceedings of the 4th International Symposium on Spatial Accuracy Assessment in National Resources and Environmental Sciences*, pages 655–658, 2000.

[180] B. Smith. Mereotopology: A theory of parts and boundaries. *Data and Knowledge Engineering*, 20(3):287–303, 1996.

[181] B. Smith and D. Mark. Ontology and geographic kinds. In T. K. Poiker and N. Chrisman, editors, *Proceedings of the 8th International Symposium on Spatial Data Handling*, pages 308–320, Vancouver, 1998.

[182] B. Smith and D. Mark. Geographical categories: an ontological investigation. *International Journal of Geographical Inforamtion Science*, 15(7): 591–612, 2001.

[183] M. J. D. Smith, M. F. Goodchild, and P. Longley. *Geospatial analysis: a comprehensive guide to principles, techniques and software tools*. Troubador Publishing Ltd, 2006.

[184] SSC. *Topographic Maps - Map Graphics and Generalization*, volume 17 of *SSC Publication Series*. SSC - Swiss Society of Cartography, Wabern, Switzerland, 2005. URL `http://www.cartography.ch/publikationen/publications.html`.

[185] S. Steiniger. *Enabling Pattern-Aware Automated Map Generalization*. PhD thesis, Department of Geography, University of Zurich, Switzerland, 2007.

[186] S. Steiniger, T. Lange, D. Burghardt, and R. Weibel. An approach for the classification of urban building structures based on discriminant analysis techniques. *Transactions in GIS*, 12(1):31–59, 2008.

[187] S. Steiniger, P. Taillandier, and R. Weibel. Utilising urban context recognition and machine learning to improve the generalisation of buildings. *International Journal of Geographical Information Science*, 24 (2):253–282, 2010.

[188] J. Stoter. Generalisation within NMAs in the 21st century. In *Proceedings of the 22nd International Cartographic Conference*, A Coruña, 2005.

[189] J. Stoter. Generalisation: the gap between research and practice. In *The 8th ICA Workshop on Generalisation and Multiple Representation*, page 10, 2005.

[190] J. Stoter. *State-of-the-art of automated generalisation in commercial software*. Number 58. Official Publication - EuroSDR, 2010.

[191] J. Stoter, K.-H. Anders, B. Baella, D. Burghardt, F. Dávila, C. Duchêne, M. Pla, N. Regnauld, P. Rosenstand, S. Schmid, G. Touya, and H. Uitermark. A study on the state-of-the-art in automated map generalisation implemented in commercial out-of-the-box software. In *The 11th ICA Workshop on Generalisation and Multiple Representation*, page 16, Montpellier, 20–21 June 2008.

[192] J. Stoter, D. Burghardt, C. Duchêne, B. Baella, N. Bakker, C. Blok, M. Pla, N. Regnauld, G. Touya, and S. Schmid. Methodology for evaluating automated map generalization in commercial software. *Computers, Environment and Urban Systems*, 33(5):311–324, 2009.

[193] J. Stoter, J. van Smaalen, N. Bakker, and P. Hardy. Specifying map requirements for automated generalization of topographic data. *Cartographic Journal, The*, 46(3):214–227, August 2009.

[194] J. Stoter, T. Visser, P. van Oosterom, W. Quak, and N. Bakker. A semantic-rich multi-scale information model for topography. *International Journal of Geographical Information Science*, 25(5):739–763, 2011.

[195] J. E. Stoter, J. M. Morales, R. L. G. Lemmens, B. M. Meijers, P. van Oosterom, C. W. Quak, H. T. Uitermark, and L. van den Brink. A data model for multi-scale topographical data. In A. Ruas, C. Gold, W. Cartwright, G. Gartner, L. Meng, and M. P. Peterson, editors, *Headway in Spatial Data Handling*, Lecture Notes in Geoinformation and Cartography, pages 233–254. Springer, 2008.

[196] R. Thomson and R. Brooks. Generalisation of geographical networks. In W. A. Mackaness, A. Ruas, and L. T. Sarjakoski, editors, *Generalisation of Geographic Information: Cartographic Modelling and Application*,

pages 255–267. Elsevier, 2007.

[197] R. C. Thomson and R. Brooks. Exploiting perceptual grouping for map analysis, understanding and generalization: The case of road and river networks. In D. Blostein and Y.-B. Kwon, editors, *Graphics Recognition Algorithms and Applications (GREC 2002)*, volume 2390 of *LNCS*, pages 148–157. Springer, 2002.

[198] R. C. Thomson and D. E. Richardson. The 'good continuation' principle of perceptual organisation applied to the generalisation of road networks. In C. P. Keller, editor, *Proceedings of the 19th International Cartographic Conference*, pages 1215–1223, Ottawa, 1999. ICA.

[199] S. Timpf. *Hierarchical structures in map series*. PhD thesis, Technical University of Vienna, Austria, 1998.

[200] S. Timpf, G. S. Volta, A. U. Frank, and M. J. Egenhofer. A conceptual model of wayfinding using multiple levels of abstraction. In A. Frank, I. Campari, and U. Formentini, editors, *Theories and Methods of Spatio-Temporal Reasoning in Geographic Space*, volume 639 of *Lecture Notes in Computer Science*, pages 348–367. Springer-Verlag, Pisa, Italy, 1992.

[201] F. Töpfer and W. Pillewizer. The principles of selection. *Cartographic Journal, The*, 3(1):10–16, 1966.

[202] A. Tversky. Features of similarity. *Psychological Review*, 84:327–352, 1977.

[203] H. T. Uitermark. *Ontology-Based Geographic Data Set Integration*. PhD thesis, University of Twente, the Netherlands, 2001.

[204] H. T. Uitermark, P. J. van Oosterom, N. J. Mars, and M. Molenaar. Ontology-based integration of topographic data sets. *International Journal of Applied Earth Observation and Geoinformation*, 7(2):97–106, August 2005.

[205] G. v. Gösseln and M. Sester. Integration of geoscientific data sets and the german digital map using a matching approach. In *Proceedings of the XXth International Society for Photogrammetry and Remote Sensing Congress*, pages 1249–1254, Istanbul, Turkey, 2004. ISPRS.

[206] P. Van der Poorten and C. Jones. Characterisation and generalisation of cartographic lines using delaunay triangulation. *International Journal of Geographical Inforamtion Science*, 16(8):773–794, 2002.

[207] P. Van der Poorten, S. Zhou, and C. Jones. Topologically-consistent map generalisation procedures and multi-scale spatial databases. In *GIScience 2002*, volume 2478 of *Lecture Notes in Computer Science*, pages 209–227. Springer, 2002.

[208] P. van Oort. *Spatial data quality: from description to application*. PhD thesis, Wageningen University, 2005.

[209] J. van Smaalen. *Automated aggregation of geographic objects : a new approach to the conceptual generalisation of geographic databases*. PhD thesis, Wageningen University, the Netherlands, 2003.

[210] A. C. Varzi. Parts, wholes, and part-whole relations: The prospects of mereotopology. *Data and Knowledge Engineering*, 20(3):259–286, 1996.

[211] R. C. Veltkamp and M. Hagedoorn. State-of-the-art in shape matching. Technical report, Department of Information and Computing Sciences, Utrecht University, 1999.

[212] V. Walter and D. Fritsch. Matching spatial datasets: a statistical approach. *International Journal of Geographical Inforamtion Science*, 13 (5):445–473, 1999.

[213] Z. Wang and J. C. Müller. Line generalization based on analysis of shape characteristics. *Cartography and Geographic Information Science*, 25(1):3–15, 1998.

[214] J. M. Ware and C. B. Jones. Conflict reduction in map generalization using iterative improvement. *Geoinformatica*, 2(4):383–407, Dec. 1998.

[215] J. M. Ware, C. B. Jones, and N. Thomas. Automated map generalisation with multiple operators: A simulated annealing approach. *International Journal of Geographical Inforamtion Science*, 17(8):743–769, 2003.

[216] J. Warmer and A. Kleppe. *The Object Constraint Language: Getting Your Models Ready for MDA*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2 edition, 2003.

[217] R. Weibel. Amplified intelligence and rule-based systems. In B. P. Buttenfield and R. B. McMaster, editors, *Map Generalization: Making Rules for Knowledge Representation*, pages 172–186. London: Longman, 1991.

[218] R. Weibel. Three essential building blocks for automated generalization. In *GIS and Generalization: Methodology and Practice*, volume 1 of *Gisdata*, pages 56–69. Taylor & Francis, London, 1995.

[219] R. Weibel. A typology of constraints to line simplification. In M. J. Kraak and M. Molenaar, editors, *Advances in GIS Research II (SDH'96)*, pages 533–546. Taylor & Francis, Delft, 1996.

[220] R. Weibel and G. Dutton. Constraint-based automated map generalization. In T. K. Poiker and N. Chrisman, editors, *Proceedings 8th International Symposium on Spatial Data Handling*, pages 214–224. IGU - Geographic Information Science Study Group, Vancouver, 1998.

[221] R. Weibel and G. Dutton. Generalising spatial data and dealing with multiple representations. In P. A. Longley, M. F. Goodchild, D. J. Maquire, and D. W. Rhind, editors, *Geographical Information Systems:*

Principles, Techniques, Management and Applications, pages 125–155. Geoinformation International, Cambridge, 2nd edition, 1999.

[222] R. Weibel, S. Keller, and T. Reichenbacher. Overcoming the knowledge acquisition bottleneck in map generalization: The role of interactive systems and computational intelligence. In A. Frank and W. Kuhn, editors, *COSIT'95: Spatial Information Theory, A Theoretical Basis for GIS*, volume 988 of *Lecture Notes in Computer Science*, pages 139–156. Springer Verlag, Berlin, 1995.

[223] E. Wentz. Shape analysis in gis. In *Proceedings of AUTO-CARTO 13*, pages 204–213, 1997.

[224] S. Werder, B. Kieler, and M. Sester. Semi-automatic interpretation of buildings and settlement areas in user-generated spatial data. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 330–339, San Jose CA, USA, 2010.

[225] M. Wertheimer. Laws of organization in perceptual forms. In W. D. Ellis, editor, *A Source Book of Gestalt Psychology*, pages 71–88. Routledge & Kegan Paul, 1923. URL `http://psy.ed.asu.edu/~classics/Wertheimer/Forms/forms.htm`.

[226] R. E. Wilson and K. M. Filbert. Crime mapping and analysis. In S. Shekhar and H. Xiong, editors, *Encyclopedia of GIS*, pages 180–186. Springer, 2008.

[227] M. F. Worboys. Nearness relations in environmental space. *International Journal of Geographical Inforamtion Science*, 15(7):633–651, 2001.

[228] F. Xie and D. Levinson. Measuring the structure of road networks. *Geographical Analysis*, 39:336–356, 2007.

[229] B. Yang, X. Luan, and Q. Li. An adaptive method for identifying the spatial patterns in road networks. *Computers, Environment and Urban Systems*, 34(1):40–48, 2010.

[230] C. T. Zahn. Graph-theoretical methods for detecting and describing gestalt clusters. *Computers, IEEE Transactions on*, C-20(1):68–86, Jan. 1971. ISSN 0018-9340.

[231] M. Zhang and L. Meng. An iterative road-matching approach for the integration of postal data. *Computers, Environment and Urban Systems*, 31:597–615, 2007.

[232] Q. Zhang. Modeling structure and patterns in road network generalization. In *Proceedings of 7th ICA workshop on generalization and multiple representation*, page 8, Leicester, 2004. (CD-ROM).

[233] Q. Zhang. Drainage typification based on dendritic decomposition. *Cartographic Journal, The*, 44(4):321–328, 2007.

[234] X. Zhang, T. Ai, J. Stoter, M.-J. Kraak, and M. Molenaar. Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. *GeoInformatica*, page 33. doi: 10.1007/s10707-011-0146-3.

[235] X. Zhang, T. Ai, and J. Stoter. The evaluation of spatial distribution density in map generalization. In *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Congress 2008)*, volume XXXVII. Part B2., pages 181–188, Beijing, 2008. URL `http://www.isprs.org/proceedings/XXXVII/congress/2_pdf/2_WG-II-2/03.pdf`.

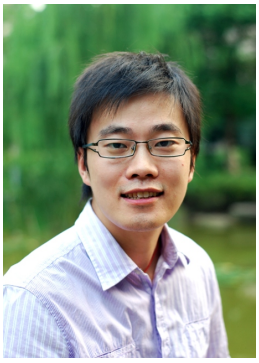
[236] X. Zhang, J. Stoter, and T. Ai. Formalization and automatic interpretation of map requirements. In *The 11th ICA Workshop on Generalisation and Multiple Representation*, page 12, Montpellier, France, 2008. URL `http://aci.ign.fr/montpellier2008/papers/24_Zhang_et_al.pdf`.

[237] X. Zhang, T. Ai, and J. E. Stoter. A voronoi-like model of spatial autocorrelation for characterizing spatial patterns in vector data. In *Proceedings of the 2009 Sixth International Symposium on Voronoi Diagrams (ISVD'09)*, pages 118–126, Washington, DC, USA, 2009. IEEE Computer Society.

[238] X. Zhang, T. Ai, and J. Stoter. Characterization and detection of building patterns in cartographic data: Two algorithms. In *Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science (SDH' 2010)*, volume XXXVIII, Part 2, pages 261–266, 2010.

[239] X. Zhang, J. Stoter, T. Ai, and M.-J. Kraak. Formalization and data enrichment for automated evaluation of building pattern preservation. In *Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science (SDH' 2010)*, volume XXXVIII, Part 2, pages 267–272, 2010.

Author's Biography



Zhang, Xiang was born on the 26th of November 1982 in Hubei province, China. He studied at the School of Resource and Environmental Sciences, Wuhan University, China from 2001 to 2005, where he obtained his Bachelor degree on geoinformatics and cartography. Later he continued his study towards an M.Sc. on map generalization and multi-scale representations at Wuhan University. Since 2008 he has carried out a doctoral study at Faculty of Geo-Information Science and Earth Observation (ITC), University of Twente, the Netherlands. His Ph.D. topic is on the quality issues in map generalization and multi-scale data. His research interest includes multi-scale data modeling, knowledge formalization, and application of computational geometry and pattern recognition in automated map generalization and evaluation.

ITC dissertations

A complete list of ITC dissertations is online on the ITC website: www.itc.nl/research/phd/phd_graduates.aspx.

This dissertation has number 213.