Institut für Photogrammetrie und Kartographie

Recognition of 3D settlement structure for generalization

Jagdish Lal Raheja

Vollständiger Abdruck
der von der Fakultät für Bauingenieur– und Vermessungswesen
der Technischen Universität München
zur Erlangung des akademischen Grades eines
Doktor Ingenieurs (Dr. Ing.)
genehmigten Dissertation.

Vorsitzender: Univ. Prof. Dr.-Ing. U. Stilla

Prüfer der Dissertation:

1. Univ. Prof. Dr.-Ing. L. Meng
2. Univ. Prof. Dr.-Ing. H. Mayer,
   Universität der Bundeswehr München

Die Dissertation wurde am 16.11.2004 bei der Technischen Universität München eingereicht
und durch die Fakultät für Bauingenieur und Vermessungswesen am
05.01.2005 angenommen.

Abstract

When a given 3D city model, consisting of mainly buildings and roads, is viewed at reduced scale, its objects not only become smaller but tend to conflict due to small area available. Generalization plays an important role to overcome these effects and helps in preserving the required legibility. Various operations are performed during generalization on these objects. One of the primary requirements for generalization is structure recognition. It not only involves structure recognition of individual object but various objects in neighborhood as well and includes many spatial relations among them. Although importance of structure recognition for 2D generalization has been reported in literature but little has been studied and reported for 3D structure recognition. This work aims at recognizing 3D settlement structures for automatic generalization, an innovative extension to 2D. The recognition procedure has been divided into three levels namely micro, meso and macro and is based upon individual buildings, buildings in neighborhood and buildings at cluster level having similar properties such as settlement blocks as well as psychophysically perceived groups. Any of these three levels of structure recognition demands that comprehensive information about the buildings should be known a-priori. Therefore first of all different buildings are recognized from the data available using a bottom-up approach. It starts with recognizing ground plans of buildings and which in turn, along with other information, are used to recognize different roof types and finally entire buildings are recognized in a similar way. After building recognition, their structure description has been studied in detail, which gives rise to various measurable parameters of individual as well as buildings in neighborhood. These parameters not only characterize individual building but also many spatial relations among them. Structure recognition at clustered level is studied next and it involves the recognition of group of buildings as a whole. The human visual system can detect many clusters of patterns and significant arrangements of image elements. Perceptual grouping refers to the human visual ability to extract significant image relations from lower-level primitive image features without any knowledge of the image content and group them to obtain meaningful higher-level structure. Various perceptual grouping principles have been applied to identify these clusters of groups. After a comprehensive study of structure recognition, their findings are then applied to 3D generalization. Among the various generalization algorithms such as aggregation, displacement, simplification, exaggeration, typification, aggregation is chosen here as it almost uses most of the results from structure recognition. Various constraints resulting from spatial relations have been already found in 2D aggregation. However, unlike in 2D, where there is only one view, the third dimension leads to many additional views and these different views become the source of additional conflicts. Apart from various views, color, texture and other small parts (window, chimney, balcony etc.) of the building also add to the existing constraints. Various additional rules have been obtained based upon these constraints. These rules along with the results of structure recognition have been used to trigger the aggregation operation. Finally, a conclusion has been drawn from the observations and the results of structure recognition and its importance to generalization. Further, its various shortcomings, problems encountered during research and future work have also been highlighted towards the end of thesis.

Zusammenfassung

Wird ein dreidimensionales Stadtmodell, das vorwiegend Gebäude und Straßen enthält in einem kleineren Maßstab betrachtet, so werden die Objekte nicht nur kleiner, sondern sie geraten aufgrund des geringen Platzangebots in Konflikt miteinander. Die Generalisierung spielt eine wichtige Rolle, um diese Probleme zu überwinden und die notwendige Lesbarkeit zu erhalten. Verschiedene Operationen werden während der Generalisierung dieser Objekte durchgeführt. Eine der primären Anforderungen an die Generalisierung ist die Strukturerkennung. Dies betrifft nicht nur die Erkennung von Strukturen einzelner Objekte, sondern auch mehrerer benachbarter Objekte und beinhaltet viele räumliche Beziehungen zwischen ihnen. Obwohl über die Bedeutung der Strukturerkennung für die zweidimensionale Generalisierung in der Literatur berichtet worden ist, gibt es bisher nur wenig Forschungen und Publikationen über die dreidimensionale Strukturerkennung. Das Ziel dieser Arbeit ist ein Beitrag zur Erkennung dreidimensionaler Siedlungsstrukturen für die automatische Generalisierung, eine innovative Erweiterung der zweidimensionalen Methoden. Der Ablauf der Erkennung wurde in die drei Stufen mikro, meso und makro unterteilt und bezieht sich auf einzelne Gebäude, Gebäudekomplexe und Gebäudeanhäufungen, welche ähnliche Eigenschaften aufweisen, wie zum Beispiel Siedliungsblöcke sowie psycho-physische menschliche Beobachtungen. Das Erkennungsverfahren wurde in die drei Stufen mikro, meso und makro unterteilt und basiert auf dem individuellen Gebäude, benachbarten Gebäuden und Gebäudeanhäufungen, die ähnliche Eigenschaften haben wie Siedlungsblöcke und psychophysische Beobachtungen von Menschen. Jede dieser drei Stufen der Strukturerkennung setzt umfangreiche Informationen über die Gebäude voraus. Deshalb werden zuerst mit Hilfe eines Bottom-up-Ansatzes unterschiedliche Gebäude aus den vorhandenen Daten identifiziert. Es beginnt mit der Erkennung von Gebäudegrundrissen, die der Reihe nach zusammen mit weiteren Informationen dafür verwendet werden verschiedene Dachtypen zu erkennen und schließlich in ähnlicher Form ganze Gebäude. Im Anschluss an die Erkennung der Gebäude wurde deren Strukturbeschreibung im Detail untersucht, was zu einer Vielzahl messbarer Parameter individueller Gebäude, wie auch benachbarter Gebäude führte. Diese Parameter charakterisieren nicht nur einzelne Gebäude, sondern auch viele räumliche Beziehungen zwischen ihnen. Als nächstes wird die Strukturerkennung von Gebäudeanhäufungen untersucht, die die Erkennung von Häusergruppen als ganzes beinhaltet. Das visuelle System eines Menschen kann viele Häufungen von Mustern und signifikante Anordnungen von Bildelementen detektieren. Perzeptuelles Gruppieren bezieht sich auf die menschliche visuelle Fähigkeit aus, auf einem niedrigen Level befindlichen, primitiven Bildmerkmalen ohne jegliches Vorwissen über den Bildinhalt signifikante Bildbeziehungen zu extrahieren und diese zu gruppieren, um aussagekräftige Strukturen auf einem höheren Level zu erhalten. Verschiedene perzeptuelle Gruppierungsprinzipien wurden zur Identifizierung dieser Gruppenanhäufungen angewendet. Nach einer umfassenden Untersuchung der Strukturerkennung, wurden die Ergebnisse in der dreidimensionalen Generalisierung angewendet. Unter den verschiedenen Generalisierungsalgorithmen wie Aggregierung, Verschiebung, Vereinfachung, Vergrößerung und Typisierung wurde hier die Aggregierung ausgewählt, da sie am umfangreichsten Ergebnisse aus der Strukturerkennung verwendet. Viele Einschränkungen, die aus den räumlichen Beziehungen resultieren sind schon bei der zweidimensionalen Aggregierung gefunden worden. Im Gegensatz zum zweidimensionalen Raum jedoch, in dem es nur eine Betrachtungsebene gibt, führt die dritte Dimension zu vielen zusätzlichen Betrachtungsebenen, die den Ausgangspunkt für weitere Konflikte bilden. Außer durch die verschiedenen Betrachtungsebenen werden die bestehenden Einschränkungen noch durch Farbe, Textur und andere kleine Teile der Gebäude (Fenster, Rauchfang, Balkon, etc.) erweitert. Basierend auf diesen Einschränkungen sind zusätzlich verschiedene Regeln festgelegt worden. Diese Regeln und Ergebnisse der Strukturerkennung wurden zur Aktivierung der Aggregierungsoperation an einem Beispiel verwendet. Am Ende wurde aus den Beobachtungen und Ergebnissen der Strukturerkennung und ihrer Bedeutung für die Generalisierung ein Schluss gezogen. Zusätzlich wurden gegen Ende der Dissertation die unterschiedlichen Nachteile und Probleme, die während der Forschung aufkamen und die zukünftige Arbeit hervorgehoben.

Acknowledgements

Without the patient encouragement and support of the following persons, this thesis could not have been realized.

First and foremost I would like to express my sincere gratitude and deep appreciation to Univ. Professor Dr. Ing. **Liqiu Meng,** for her excellent supervision on this research and her great advice in the preparation of this thesis. Her valuable guidance, inspiring discussion, stimulating suggestions and constructive criticism have been of great importance and influence to me, both academically and personally.

I would like to take this opportunity to thank President, Technical University München for giving me an opportunity and Director, CEERI, PILANI (INDIA) for allowing me to pursue my research. This research was sponsored by Deutsche Forschungsgemeinschaft (DFG), Bonn under the project „Generalisierung von dreidimensionalen (3D) Siedlungsstrukturen auf Grundlage von Maßstabs-räumen und Strukturerkennung".

I am grateful to Prof. Dr. Ing. **Helmut Mayer,** Universität der Bundeswehr München, for his constant encouragement and helpful suggestion.

My sincere thanks are also due for Ms **Adrea Forberg**, Universität der Bundeswehr München, for her constant interaction and suggestion throughout the research.

The cooperation I received from other faculty members of this department is gratefully acknowledged. Dr. **Tumasch Reichenbacher** helped me to understand the basics by providing relevant literature. Mr **Fritz Meier** never hesitated to provide required infrastructure and Mr **Robert Kauper** provided all technical help promptly whenever I needed. Mr **Angsüsser Stephan** always & promptly helped me whenever I approached him for any problem and also improved my German language. Ms **Daniela Töllner** is my JAVA *Guru* as she helped me in resolving many JAVA problems during research related software development. Taste of coffee will never be same again as it was during coffee break and the coffee is always prepared by Mr **Theo Geiss**. To be fit, I stole some time from the research to play TT with Mr **Anton Siebenlechner.** I will be failing in my duty if I do not acknowledge Mr **Gabriele Kursell** and Mr **Aloisia Fleißner** for their timely administrative help.

I wish I would never forget the company I had from my fellow research scholars of my department. In particular, I am thankful to Mr. **Rakesh Kasturi**, for his great patient, in proof reading, Dr **Lichun Sui**, Ms **Ming Chen**, Ms **Jie Shen,** Mr **Oliver Swienty,** Dr **Shi Wei** and Mr **Abel Sebastian** for their help and support.

I also want to thank my family, who taught me the value of hard work by their own examples. My lovely daughter, **Sonia**, despite passing through crucial & deciding period of her career and where my presence and guidance was must for her, encouraged me continue my doctorate. I would have never thought a better daughter than her. I would like to share this moment of happiness with my wife **Gulshan** and son **Shekhar**. They rendered me enormous support during the whole tenure of my research. The encouragement and motivation that was given to me to carry out my research work by my **parents** is also unforgettable.

List of Figures

List of Tables

Table of Contents

Chapter 1

Introduction and objectives of the work

1.1 Introduction

Humans perceive their environments primarily in terms of visual variables. What makes an object similar or different from another object is its structure. Most of these objects are made of many parts or primitives and therefore an object's structure can be characterized by relationships between these primitives. A structure is an entity made up of a number of parts having inter-relationship among them that are held together in a particular way and has a base or foundation. Its shape or size remains fixed and can change only if the material it is made from can bend or stretch. These structures can be natural as well as manufactured. Natural structure include human beings, trees, plants and formations in rocks, such as caves and manufactured structures which are made either in a workshop, on a production line or assembled on site. A 3D city model consists mostly of buildings and roads. Each object is a 3D structure, which in turn consists of geometric and topological data. When these structures are represented geometrically, it involves a graph having nodes, edges and associated numerical quantities (such as a numerical value labeling the nodes and edges). Moreover, there are consistency constraints governing the relation between the combinatorial structure and numerical quantities. This means that perturbing the numerical values without taking into account the combinatorial structure can lead to qualitatively different or inconsistent structure.

Structure recognition of 3D graphical objects is an important problem that can be hardly solved without intelligence. It is the identification of these objects and aggregate objects, their spatial and semantic relations and their relative importance. The importance of these spatial relations for the recognition of structures embedded in a complex environment cannot be ignored. Structure recognition has been applied to various fields such as computational biology where protein's unique 3D structure is recognized (Boris 1997), cartographic generalization, structure recognition of on-line handwritten documents (Moin 1997) and many others. Cartographic generalization of 3D city models is one of the burning problems that need further attention.

3D city models represent an abstraction of reality and are the most powerful tools to quickly provide information about places and spatial relationships to people searching for geographical information. Computer generated perspective views are often simply referred to as 3D maps. Although this term is not yet strictly found in the cartographic literature, there are specific reasons why it should be used. These views are referred as 3D, because we perceive the presented scene with our human perception system in a three-dimensional perspective way, even when the scene is depicted on two-dimensional media. And, "maps", because these products integrate and display spatially-arranged phenomena in accordance with cartographic symbolization conventions (Haeberling 1999) and (Haeberling 2003). Nevertheless, although they possess cartographic characteristics, 3D maps should be considered a map-related representation. Since 3D maps for a city are produced at large scale (~1:5000) so that individual buildings are easily recognizable and therefore in that context 3D maps may be defined as "map like representation of the large scale city model on a digital media adhering to the cartographic rules".

When 3D maps are reduced from larger scale to smaller scale, their objects find less space to be accommodated in the available area and tend to conflict with each other. In order to preserve the legibility of the maps, certain objects have to be deleted, aggregated and so on. These objects cannot be manipulated randomly and at will but a comprehensive knowledge about their structure and spatial relations with their neighbors is needed. It is here where the cartographic generalization plays an important role. It is a multi-step process based on manipulation of graphic characteristics of 3D objects. Depiction of these objects on the screen depends upon the generalization process it has passed through. Main emphasis should be the criteria that the general structure of these objects after the generalization should be preserved.

According to the conceptual framework (Brassel 1988), the overall process of map generalization consist of five steps: *Structure recognition, process recognition, process modeling, process execution, and data display*. The generalization constraints (scale reduction factor, map purpose, data quality, etc.) provide control for these tasks. The purpose of structure recognition and process recognition essentially is to determine the relevant structures and the relative importance of the map elements of the input map, and identify the generalization operators that should be used to solve the given generalization problem.

In the early 1960s, when only paper maps existed, generalization was a manual operation. Cartographers distinguished between graphic and conceptual generalization. The first would mainly deal with the geometry and the second would in addition result in a change of the legend items as well.

In the beginning of the 1980s, automatic generalization was performed only at an experimental level and required much interactivity e.g. every object needed individually to be generalized with either specific or non-specific generalization algorithms. Dealing with spatial relationships between objects was not of primary interest in this phase. Generalization was still at its semi-automatic stage.

However, during mid eighties, as the automation of generalization continued, need of structure recognition became more important and eventually inevitable. For example, which road has to be deleted or kept, highly depends upon its proximity to nearby objects. The generalization concepts illustrated in the subsequent sections underline the importance of structure recognition.

1.2 Generalization processes and their dependencies on structure recognition

Generalization processes require maintaining the overall structures and patterns presented with the source map or database. It may be regarded as a process of increasing the level of abstraction relative to the original surveyed form of the geographical features. These various processes and sequence of their application depends upon its structure and surrounding objects. Thus the recognition of the structure, mainly implicit and previously unknown, has been an important task to achieve a better generalization outcome (Jiang 2004). It takes place as soon as features or feature classes are represented on a medium for communication. Most of the important processes are described below:

I.Simplification. It involves the smoothing and elimination of small and unimportant features as shown in figure 1, parts, those are too small on the surface of a building, are removed in order to make it readable at a given scale. For example, small windows, doors and chimneys come under these details and can be removed by keeping a threshold on the size of the features. Structure recognition provides this information about the size, area and volume of various parts of the buildings and helps in deciding their removal.

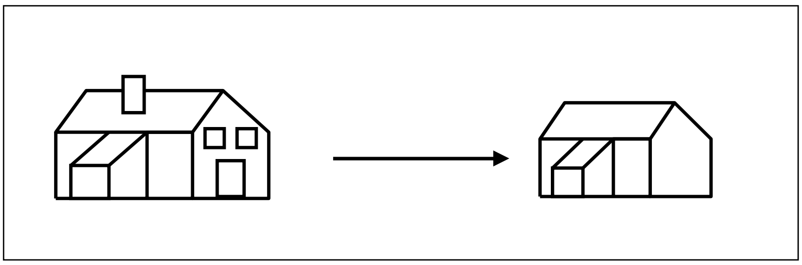


Figure 1: 3D simplification

II. *Exaggeration.* It is used to enlarge parts of objects as shown in figure 2, either because they are very small and do not satisfy the geometric constraints and because such parts are important and of special interest (Jones 1995). Structure recognition gives complete information of the part to be enlarged.

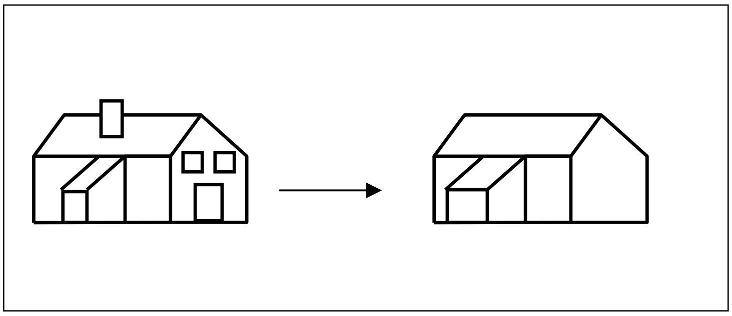


Figure 2: **Exaggeration**

III. Graphic aggregation. It involves the representation of a group of objects with another representation as shown in figure 3. It requires all the individual building details such as roof type, ground plan etc and proximity information, which can be only obtained through structure recognition.

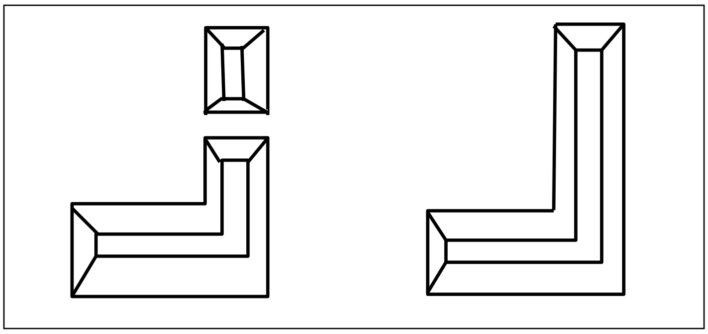


Figure 3: Aggregation

IV. Displacement is a local transformation of a set of objects in order to solve proximity conflicts as shown in figure 4. The displacement may arise because of aggregation and exaggeration operations. For example, there may be a building situated close to a road (Thick black curve in figure 4). It is desired to exaggerate road so that now it is represented by two lines (two dotted lines). While doing so, it may result in a conflict with the nearby building and therefore forcing in its displacement. Certainly local transformation of such a building is not possible without its structure recognition.

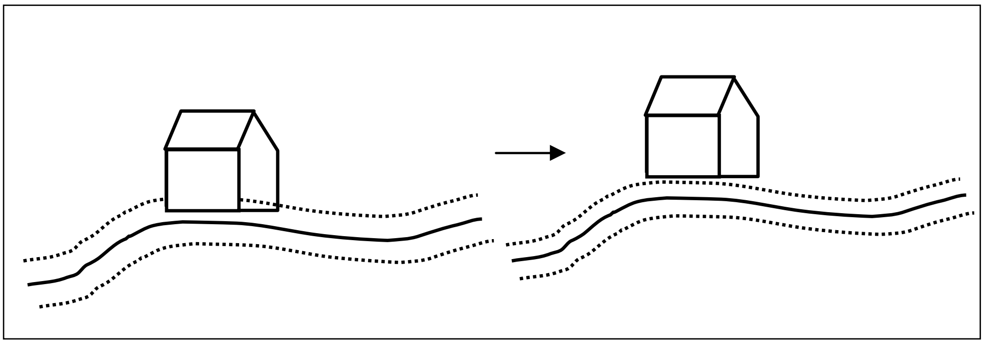


Figure 4: **Displacement**

V. Typification. The major aim of typification is to reduce feature density, form variations and the level of detail while maintaining the representative distribution pattern and visual impression of the original feature group. It involves the replacing of a large number of similar objects by a small number (Sester 2001; Anders 2003), as shown in figure 5, while ensuring that the typical spatial structure of the objects is preserved. Different parts of the city exhibit different cluster densities. These differences have to be preserved, if not enhanced, by typification. It requires not only the structure recognition of individual objects but of a group as well forming a cluster.

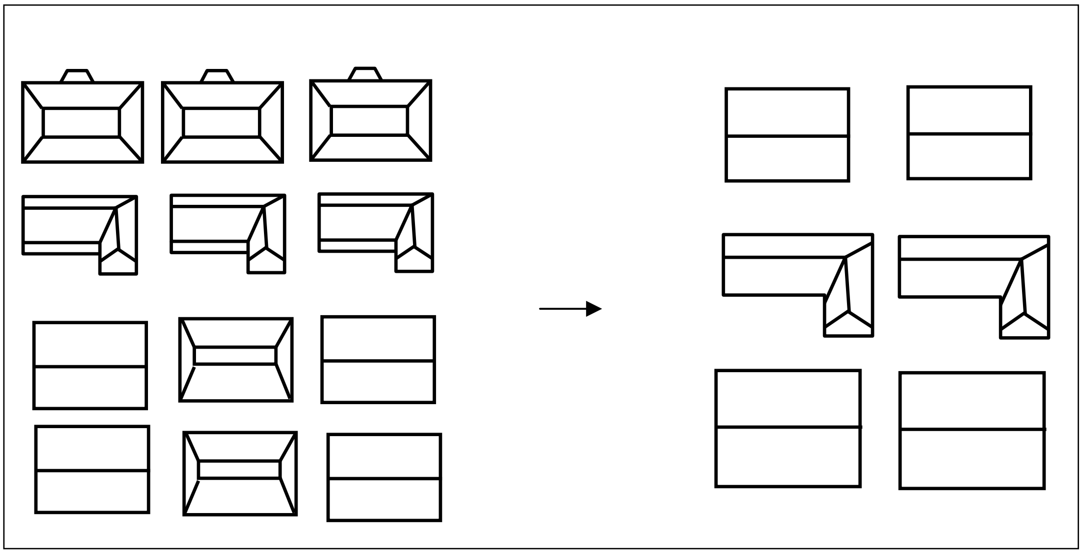


Figure 5: **Typification**

VI. Selection: When the map scale is reduced, the density of the objects increases and therefore decision should be made on which objects should be selected (Kreveld 1997), (Harrie 2002). An intellectual process and decides which classes of symbols will be necessary to serve the visualization purpose. While doing so, no modification takes place to the symbols as shown in figure 6. However, the distribution pattern must be preserved. Structure recognition again plays an important role in preserving the patterns.

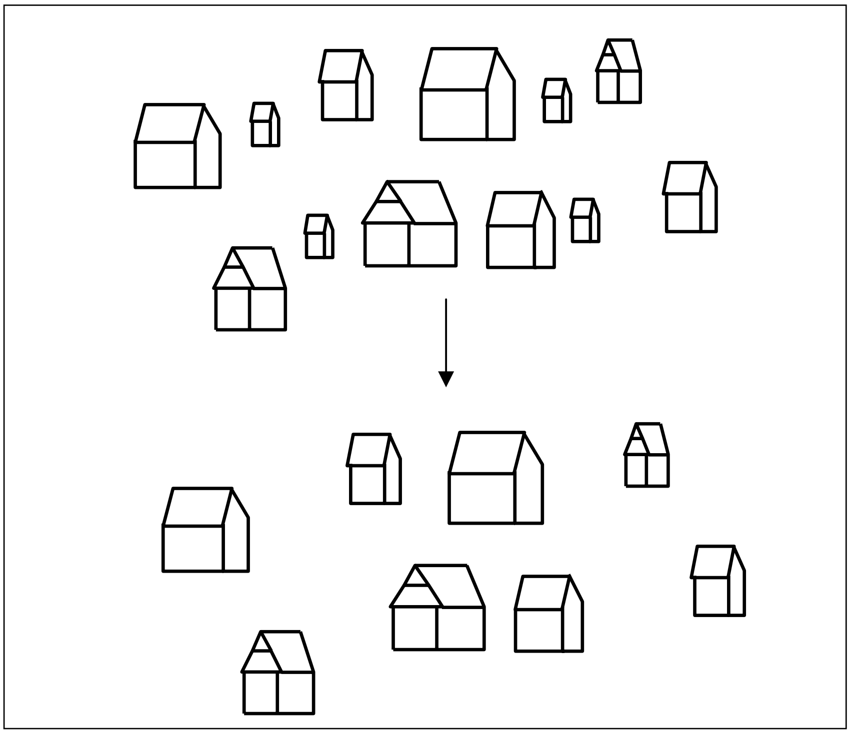


Figure 6: Selection

1.3 Need for structure recognition

These various generalization processes, discussed above, may affect the general topological structure of the objects continuously. For example, aggregation corresponds to joining a group of different features into a higher-order feature. Simplification may change the general shape of an individual line, which may cause changes in the relationship of this line with other components of the map. Similarly other process, viz. typification, displacement, selection may have the same effect. In fact, it can be viewed as a set of metric transformations on the geometric representations of spatial objects, intended to improve data legibility and understanding. Before various generalization algorithms, based upon the various processes described above, structure recognition is the most important task to be done. According to (Brasset 1998) and (McMaster 1992), it is the process of determining what to generalize, why, and when to generalize it. Brassel, Weibel, and Dutton (Brasset 1998), (Weibel 1997a) and (Weibel 1998) recommend integrating structure recognition into an overall framework to determine the generalization objectives. (McMaster 1992) used structure recognition, an aspect of what they call cartometric evaluation, to identify the typical geometric conditions that would trigger a generalization procedure,

- When crowding of various graphic objects happen, i.e. congestion.
- When some of the graphic objects seem to fuse together, i.e. coalescence
- When the graphic objects overlap each other such as a building try to overlap a nearby road. I.e. conflict.
- Complication, as aspect of special-cases, which are not easy to identify a-priori inconsistency, when objects are wrongly transformed differently under similar cartographic conditions

Since the beginning of the era of digital generalization, efforts have been made to develop algorithms for various aspects of generalization. During the first period of generalization (1960-1975) these efforts were devoted to a rather narrow aspect of the overall problem (Weibel 1991) such as line simplification, selection and displacement of point and line symbols. Nevertheless, research on generalization needs to go beyond the development of these geometric operators and effort should be made to develop algorithms, which incorporate context information (an object is not generalized in isolation, but in relation to other objects). During 1980s, however the considerations began to move towards the more conceptual aspects and structure recognition was an important concept among them. They addressed the need that generalization algorithm should incorporate the additional constraints to preserve the general structure of the objects and to keep them consistent at different levels. (Monmonier 1989) stated “the lack of efforts in generalization in computer- assisted cartography stems not only from the computational complexity of the problem but also from only a vague understanding of the objectives and principles of map simplification”. He regarded the automatic recognition of essential feature characteristics as the major obstacle to fully automated cartographic generalization.

Structure recognition has the primary goal to extract, quantify, and formalize the explicit and implicit knowledge embedded in a 3D structure such as 3D buildings, among buildings in spatial or semantic proximity, within a settlement block or an apparent distribution pattern. Only when the spatial characteristics of settlement structures are captured and made available for computer programs, it is possible to develop a comprehensive generalization system. It will contain not only a set of generalization operations, but also necessary constraints and rules. Further, it will help for the automatic detection of instances for generalization, determination of required generalization operations as well as their access sequences, and fine-tuning of the valid parameter ranges in combination with the iteration degree of each individual operation. Another goal of structure recognition lies in the application of its results as an objective indicator of generalization quality.

The information gained from this study will help in formulating the constraints and rules for 3D generalization. It will help us in deciding what operator is needed for an object or a group of objects.

Since we know, beforehand, each and every detail (e.g. type, topological, semantic and contextual information) of the objects being generalized, it will facilitate in great way to pursue generalization and consequently give better results.

1.4 Approach

Brassel and Weibel (Brassel 1988) proposed a conceptual framework that identifies the major steps of the manual generalization process and transposes these concepts into the digital realm. It contains the following approach:

- Structure recognition: This process aims at the identification of objects and aggregate objects, their spatial and semantic relations and their relative importance.
- Process recognition: The second step is to define the relevant generalization processes. This involves the identification of the types of data modifications and the parameters controlling these procedures.
- Process modeling: Next step is called *process modeling*, which compiles rules and procedures from a process library.
- Process execution: Digital generalization takes place under this step, where the rules and procedures are applied to the source database in order to create the generalized target database.
- Data display: The last process converts the target database into a fully symbolized target map.

It is a complex research issue that has existed as a design problem ever since the first maps were made. Lots of work has been done (Meng 1997), (Meng 1998) on 2D generalization but comparatively very less has been reported for 3D generalization. A framework has been proposed in this dissertation that highlights the issue related to 3D generalization based upon structure recognition of buildings. At first, structure description of individual buildings, buildings in neighborhood and buildings forming a cluster is studied. A different set of parameters is required at each level. Structure recognition of individual buildings, simple as well as complex, is discussed in great details using Artificial Neural Network techniques. It is followed by structure recognition of group of buildings forming a cluster exhibiting perceptual grouping behavior. A set of new rules and constraints are developed based upon structure recognition and consequently algorithms are implemented for aggregation.

1.5 Proposed research

The first and primary goal of this research is the recognition of 3D settlement structures using a hierarchical approach. This step aims at the identification of various building types, their spatial and semantic relations, and their relative importance. A new approach using Artificial Neural Network (ANN) is applied for the identification of various kinds of 3D buildings. Based on this recognition process and keeping in mind the work done for 2D generalization, new rules and constraint are derived which form the basis for new algorithm for 3D aggregation.

1.6 Organization of the thesis

In the next chapter, the background to the problem at hand is reviewed. Various method applied in structure recognition have been analyzed along with their drawbacks. Its importance to generalization, covering one, two and three-dimensional, is also discussed by reviewing the pertinent literature.

Structure recognition is introduced in chapter 3 by categorizing different buildings based upon their roof style and shapes. Various parameters are needed to define them before proceeding to recognize their structure. Structure description of buildings is divided into three parts, viz: micro, meso and macro levels, and each level is discussed in details.

Chapter 4 gives an overview of the Artificial Neural Networks that are applied to structure recognition of buildings.

In chapter 5, the implementation of the hierarchical structure recognition of 3D building, using Artificial Neural Network (ANN) approach, is addressed. A comprehensive methodology is discussed and explained. The whole process is completed using real 3D data of an area of the city of BONN, Germany, comprising different buildings.

In chapter 6, structure recognition of group of buildings forming a cluster is studied. These clusters are formed based upon perceptual grouping mechanisms.

In chapter 7, different constraints and rules are developed based upon the results of structure recognition obtained in chapter 5 and 6 and consequently aggregation algorithm has been developed and implemented. Its results are also presented and discussed here.

Chapter 8 first summarizes what has been done in this research. The results and the conclusions of our investigations are stated and their limitations are discussed. Finally, new ideas on how to continue further research on this topic are given.

Chapter 2

The State of the Art of structure recognition

2.1 The State of the Art of structure recognition

Understanding how biological visual systems recognize an object is one of the ultimate goals in computer science and is largely based on the matching of descriptions of shapes (Riesenhuber 2000). Literature survey has revealed that a lot of research has been done for *object centered* structure recognition of 2D objects and less effort has been applied to the *view centered* structure recognition of 3D objects. This chapter will concentrate on some of the popular methods. Among the various methods available, shape descriptors such as Fourier descriptor and wavelet descriptor, moment invariants and Geon theory have been widely used to describe shape irrespective of position, orientation and scale. A brief review is done here by giving introduction, their applications and both advantages & disadvantages.

2.1.1 Structural description theory

Humans are able to comprehend novel images of objects and scenes, often under highly degraded and novel viewing conditions. This is achieved in a fraction of a second. It is done when human mind puts together primitives (features that allow specific recognition) in a system that breaks down objects into simple 3D shapes. This fact is the basis of *structure description theory*. Based upon this theory, Marr and Nishihara were the first to develop a reasonably complete structure-based approach to object recognition (Marr 1978). They proposed a general method for shape recognition, in which objects are represented by a set of generalized cylinders as shown in figure 7. This set of generalized cylinders is organized as a hierarchy, where smaller parts are at the lowest levels. In other words, an object can be decomposed into axes of generalized cones depending on the nature of detail required. Following three steps are involved in recognition:

- *Single-model axis*. The first step in the model is identification of the main axis of the object.
- *Component axes*: The axes of each of the smaller sub-portions of the object are identified.
- *3D model match*: Finally, a match between the arrangement of components and a stored 3D model description is performed to identify the object

Thus for example, to differentiate between a flat roof and gable roof buildings, the decomposition would result in an arrangement of axes of differing lengths representing generalized cones which in turn approximate the roof, walls and windows of the buildings. The axes of the objects are derived mainly from the occluding contour or silhouette of the image.

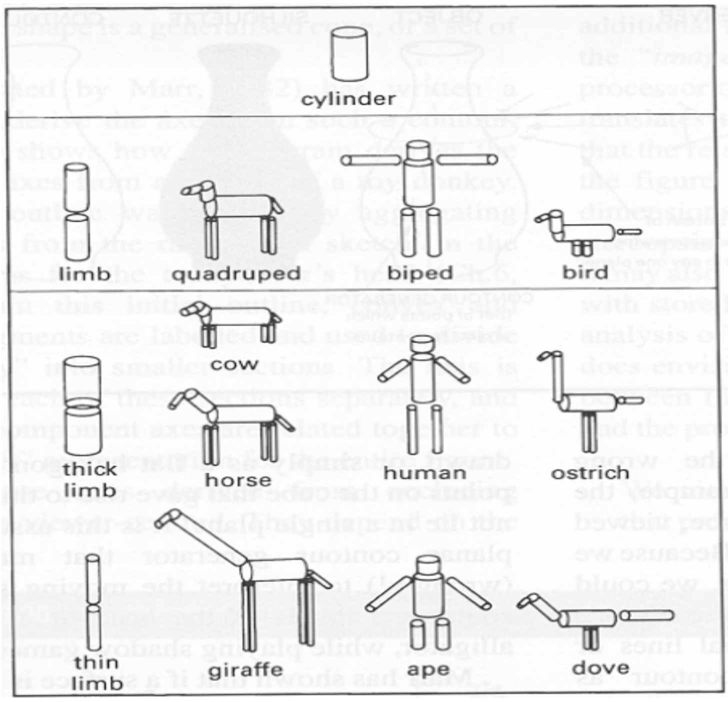


Figure 7: **The catalogue (Marr & Nishihara, 1978)**

Although object comparisons are fastest if the main axis of an object is the same as the object it is being judged against, however this technique is restricted to the set of objects that can be described as generalized cones with a clear main axis and a constant-shape cross section

2.1.2 Geon theory

Another more recent and elaborate structural object recognition theory, which allows more complex basic shapes than cylinders is, of Biederman and his co-workers (Biederman 1987). According to this theory, objects as perceived, are decomposed into 3D primitives called geons, together with the skeleton structure connecting them. The theory proposes a hierarchical set of processing stages, leading to object recognition. In the first two stages, images of objects are decomposed into edges, then into component axes, oriented blobs, and vertices. Following this, 3D primitives such as cones, cylinders and boxes are identified. A central concept in Biederman's approach is that a set of generalized cones or "geons" (short for geometrical ions) are 3D perceptual primitives. He defined a family of 36 geons by image properties of the silhouette contours in the 2D plane, by co-linearity, symmetry, parallelism, curvature, and co-termination (the contours meet at a point, e.g. a cone). The next stage extracts the structure that specifies how the geon components interconnect. The decomposition of an object results in a Geon Structural Description (GSD), consisting of geons, their attributes, and their relations with adjacent geons. It is this structural description that contributes to viewpoint invariance. Therefore if two views of an object result in a similar GSD, then they should be treated as equivalent by the object recognition system. Figure 8 illustrates a subset of geons and some simple objects constructed with geons.

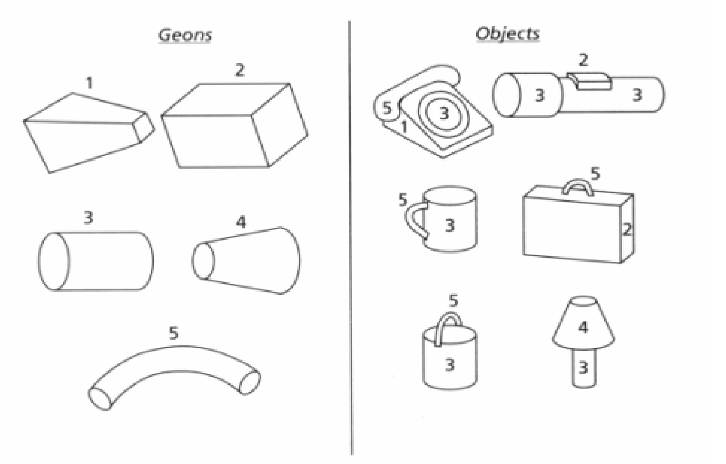


Figure 8: A set of generalized cones or "geons" (Biederman, 87)

According to geon theory, color and texture are surface properties of geons that play a secondary role in object classification. These properties may aid in the recognition process, but do not constitute the defining characteristics.

One advantage of the geon theory is that its demands on the sketch are not unreasonable. Carving objects into parts, labeling the parts as geons, and ascertaining their arrangement are not insurmountable problems, and vision researchers have developed models of how the brain might solve them. Another advantage is that a description of an object's anatomy helps the mind to think about objects, not just to blurt out their names. People understand how objects work and what they are for by analyzing the shapes and arrangements of their parts. However, Biederman's feature set is severely limited in its application to many natural objects (Schyns 1998). It doesn't allow discriminations between many similar categories, and objects within the same category will not necessarily be represented by the same geon structure. These limitations are a problem not for Biederman's theory alone but also for any approach that cannot adapt its building blocks flexibly to categorical constraints.

2.1.3 Shape descriptors

A descriptor is a representation of a feature and defines the syntax and the semantics of the feature representation. They are also commonly used for shape recognition. Following is the brief introduction of the various descriptors:

2.1.3.1 Fourier descriptor

Fourier descriptor is used for shape description. These Fourier descriptors are used to describe the boundary of a shape in 2D space using the Fourier methods.

Here one starts with taking N points digital boundary of a shape on xy-plane e.g. a facet of a 3D object. These N points may represent all the pixels occupied by the boundary or a set of sample from them. Let

$$\mathbf{A = \{P(X_k, Y_k), k=1....N\}}$$

Be the set of these points with coordinates (X,Y) represented in xy-plane. If the labels on each axis be replaced by name as horizontal axis for "real", and the vertical axis for "imaginary", then the graph consist of complex numbers.

These points can be represented as s(k) where

$$s(k) = X_k + iY_k \text{ for } k = 0,1,2,....N-1$$

Although the interpretation of the sequence has been recast, the nature of the boundary itself has not been changed. The advantage of this representation is that it reduces a 2D into a 1D problem, i.e. there are now N complex numbers instead of 2*N real numbers.

Applying discrete Fourier Transform of s(k) gives

$$a(u) = \frac{1}{N}\sum_{k=0}^{N-1} s(k)\, e^{-i2\pi ux/N} \text{ for u = 0,1,2,3,..N - 1}$$

The complex coefficients a(u) are called the Fourier descriptors of the boundary. Applying inverse Fourier transform to a(u), s(k) will be restored.

$$s(k) = \sum_{u=0}^{N-1} a(u)\, e^{i2\pi ux/N} \text{ for k = 0,1,2,3,..., N - 1}$$

However, not all coordinate values are necessary to reconstruct the original image. One can “drop” the Fourier descriptors with higher frequencies because their contribution to the image is very small. Expressing this as an equation,

$$s(k) = \sum_{u=0}^{M-1} a(u)\, e^{i2\pi ux/N} \text{ for k = 0,1,2,3,..., N - 1}$$

This is equivalent to setting a(u) = 0 for all terms where k > M-1, where M<N.

The more descriptors you use to reconstruct the original image, i.e. the larger the M, the closer the result gets to the original image. In practice, an image can be reconstructed reasonably well even though not all the descriptors are used.

The Fourier descriptor is a widely used all-purpose shape description and recognition technique (Granlund 1972) and (Winstanley 1998). The shape descriptors generated from the Fourier coefficients numerically describe shapes and are normalized to make them independent of translation, scale and rotation. These Fourier descriptor values produced by the Fourier transformation of a given image represent the shape of the object in the frequency domain (Wallace 1980). The lower frequency descriptors store the general information of the shape and the higher frequency the smaller details. Therefore, the lower frequency components of the Fourier descriptors define a rough shape of the original object

The Fourier transform theory can be applied in different ways for shape description. To apply the Fourier descriptor technique to cartographic data, the points are stored as a series of complex numbers and then processed using the Fourier transform resulting in another complex series of the same length N(Keyes 2001). If the formula for the discrete Fourier transform were directly applied each term would require N iterations to sum. As there are N terms to be calculated, the computation time would be proportional to N^2 . Therefore, the algorithm chosen to compute the Fourier descriptors was the Fast Fourier Transform (FFT) for which the computation time is proportional to *NlogN*. The FFT algorithm requires the number of points N defining the shape to be a power of two.

However, the Fourier descriptor has several obvious shortcomings in shape representation. Since the Fourier basis is not local in the spatial domain, a local variation of the shape can affect all the Fourier coefficients.

2.1.3.2 Wavelet descriptors

Wavelet descriptors are used to describe the boundary of the shape. (Pfeiffer 1995) showed that these descriptors are more suitable than Fourier descriptor for describing the boundary of the shape. (Wunch 1995) presented the wavelet descriptors in the framework of an application for the multi-resolutional recognition of handwritten characters. The features were obtained by computing the wavelet transform for the boundary of the characters. The experiment showed that wavelet descriptors work more robustly than the Fourier descriptors as far as different writing styles are concerned. It also provides better coarse shape features at the low frequencies as well as detail features at high frequencies. Besides, it offers a natural multi resolution representation of the signal so that a multi resolution matching can be employed. Disadvantage with it, however, is to normalize the input signal so that the starting point is fixed at a specific position.

The use of wavelet descriptors involves intensive computation in the matching stage as these wavelet descriptors are not rotation invariant. For example, both (Tieng 1997) and (Yang 1998) use best matching method to measure similarity between two feature vectors of the two shapes, this is impractical for higher dimensional feature matching. Therefore, wavelet descriptors are more suitable for model-based object recognition than data-driven shape retrieval, because the speed is essential for shape retrieval, which is usually conducted online.

2.1.3.3 Shape context descriptor

Shape context descriptor (Malik 2002) describes the coarse distribution of the rest of the shape with respect to a given point on the shape. Finding correspondences between two shapes is then equivalent to finding for each sample point on one shape the sample point on the other shape that has the most similar shape contexts. Once the correspondence at sample points is given, it can be extended to the complete shape by estimating an aligning transformation that maps one shape onto the other. Consider the set of vectors originating from a point to all other points on a shape. These vectors express the configuration of the entire shape relative to the reference point. If there are n points, a coarse histogram h_i of the relative coordinates of the remaining n-1 points is given by

$$hi(k) = \# \{q \neq p_i : (q - pi) \in bin(k)\}$$

Where *bin(k)* are used as they make the descriptor more sensitive to positions of nearby sample points than to those of points farther away. The above equation tells us that the shape context of an image point p_i is a histogram, which describes the relative position of the remaining points. This shape context at a reference point captures the distribution of the remaining points relative to it, thus offering a globally discriminative characterization. Corresponding points on two similar shapes will have similar shape contexts and therefore enable us to solve for correspondences as an optimal assignment problem. Given the point correspondences, a transformation that best align the two shapes can be estimated. The main problem with the shape context shape descriptor is that they are very sensitive to image distortions.

2.1.4 Moment invariants

An image of a 3D object may be described and represented by means of a set of its moments. The set of moments of an object can be normalized to be independent of the object primary characteristics, namely, translation, rotation, and scale. Hence, moments can be used to recognize 3D objects. These moments can provide characteristics of an object that uniquely represent its shape. Several techniques have been developed that derive invariant features from moments for object recognition and representation. These techniques are distinguished by their moment definitions, such as the type of data exploited and the method for deriving invariant values from the image moments. It was Hu (Hu 1962) that first set out the mathematical foundation for two dimensional moment invariants and demonstrated their applications to shape recognition. They were first applied to aircraft shapes and were shown to be quick and reliable (Dudani 1997). These moment invariant values are invariant with respect to translation, scale and rotation of the shape. Hu defines seven of these shape descriptor values computed from central moments through order three that are independent to object translation, scale and orientation. Translation invariance is achieved by computing moments that are normalized with respect to the centre of gravity so that the centre of mass of the distribution is at the origin (central moments). Size invariant moments are derived from algebraic invariants but these can be the result of a simple size normalization. From the second and third order values of the normalized central moments a set of seven invariant moments can be computed which are independent of rotation. Moment invariants have found wide application in pattern invariant recognition since it was proposed. The main difficulty in the application of moment invariants is in their computation.

2.1.5 Template matching

Template matching is the process by which the mind identifies objects in comparison to, a particular kind of, stored mental representation. According to the theory, the mind carries around a vast storehouse of images that can be compared with visual input. An object is identified by "matching" the mental image. Thus, in classic form, template matching involves comparing the input with a pictorial, global, or non-decomposed representation of objects. (Rainsford 2002) explored this idea, inspired by the Danish Mapping Agency KMS for the generalization of rural buildings, to select from a set of templates, a building outline that best characterizes a more detailed form. These rural buildings are represented using a series of simple alphabetic templates with similar shapes as shown in figure 9. The use of the templates in effect creates caricatures of the farm building outlines as shown in figure 10 that shows the original building in grey and a fitted template in outline. It requires the recognition of polygon shapes as well as the simplification procedure (to generalize).

Figure 9: The templates I,F,P,G,E,L,U,O,T [Rainsford, Mackaness, 2002]

A single building polygon is chosen and matched against a set of defined templates. These template shapes can be stretched or flattened to achieve satisfactory results.

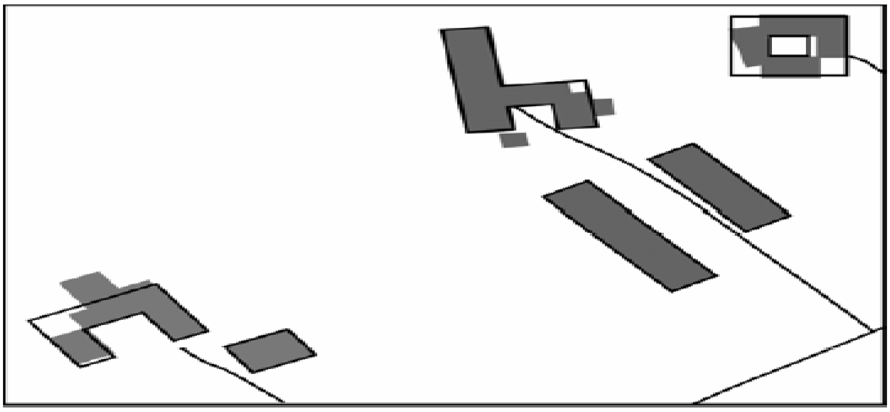


Figure 10: **Template matching [Rainsford, Mackaness, 2002]**

The template-fitting scheme first uses a two-step process of simplification followed by a template selection procedure as shown in figure 11(Rainsford 2002). Some building shapes have central courtyards or “holes” in them, by counting these holes, an appropriate template group I,L,U,T,F,E group (one ring) or O,P (2 rings) is selected. Internal angles of the simplified and squared polygons are measured and recorded in a sequence to characterize their shape. Therefore based on the number of holes in the object, the number of vertices and the sequence of internal angles, the choice of the templates is progressively narrowed down.

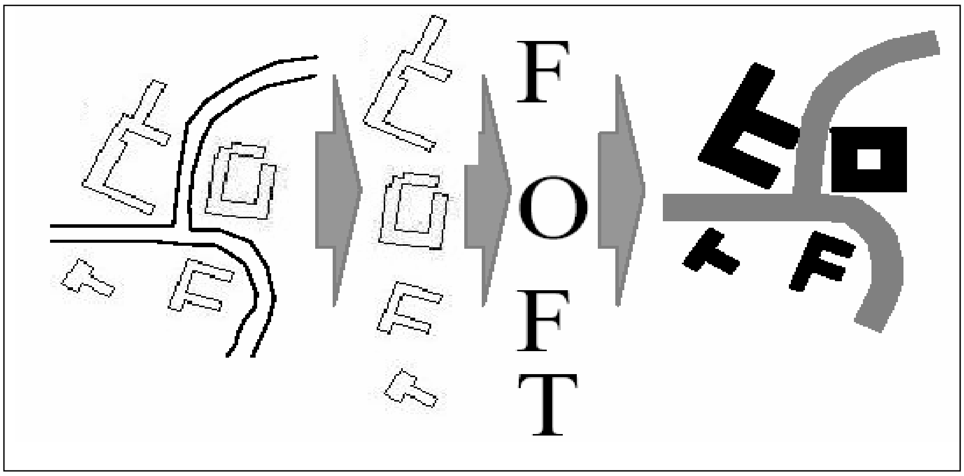


Figure 11: Template matching process [Rainsford, Mackaness, 2002]

Even though this process was used for 2D shapes, it still requires a lot of simplifications before vertices, angles and loops are measured. It fails when two near identical shapes (like E and F) are to be recognized. 3D buildings with angles less than 90^0 (especially roofs) will make the recognition more difficult as this process deals with right angles only.

2.1.6 Feature analysis

Feature analysis theories provide an alternative to the template approach. In the feature analysis approach, shapes are represented in memory as configurations of distinctive and separable parts called features (Bassett 2003). For example, a capital 'A' has a horizontal line and two diagonal lines (in opposite orientations). This is true of A's in almost all fonts and styles, even though they may differ dramatically in terms of the intensity arrays they might produce.

Feature models differ from template models in two major aspects. First, shapes are stored in memory as structural descriptions (a list of features and their relationships) instead of templates (in an intensity-array format). Second, feature models insert an additional stage of preprocessing between selection and recognition. In this stage, the intensity array in iconic memory is analyzed to identify the features in the array and note how those features are related. As a result, both the current input and the structural descriptions stored in long-term memory have the same format--lists of features and feature relationships. Thus, whereas the matching process in the template model compares two intensity arrays, the matching process in the feature model compares two feature lists.

Feature analysis has several advantageous properties. First, it provides a relatively robust solution to the problem of shape variability. Two members of a category may have many superficial differences and yet share the same underlying features. Second, feature analysis provides an intuitively (and experimentally) more plausible mechanisms for making similarity judgments. One case where this can be especially valuable involves an encounter with an unfamiliar pattern. Both the template matching and feature analysis models were designed mostly to recognize alphanumeric characters. Their suitability for 3D objects is less investigated.

2.2 Current application of structure recognition in generalization

A basic level of understanding of the structure of map objects is necessary for its successful generalization. During the cartographic generalization, geographic objects cannot be considered one by one but in relation to each other. The relations are either hierarchical or non-hierarchical. The most important *hierarchical relations*, according to (Mustiere 2002) are:

• *Part of a group*: In a map, lots of objects take a more precise meaning by being a part of a group than on their own. For example, in certain studies a highway interchange may be more significant than the isolated road sections it contains. Identification of significant groups is very important during generalization. First, some generalization processes cannot be done while looking at objects one by one and therefore must be preformed at the group level (such as merging and typification). Second, the fact of being part of a significant group can influence the way the elements of the group are generalized.

• *Being inside a particular area*: This type of spatial contexts considers the property of being inside an area that can be qualified by some global characteristics. For example, a garage is within the area of building or near to it.

• *Non-hierarchical relations*: deal with another type of spatial context those influences the generalization process based on the local relations that a given object have with its surrounding objects. For example, a building near a road, a building is aligned with a road. These relations influence the generalization process in different way.

It is therefore important to understand what type of relations may exist between objects in order to understand how they influence the generalization process thereby underlying the importance of the notion that structure recognition is an inseparable part of automatic generalization. The structure recognition along with the knowledge of scale space plays an essential role in determining the local and global constraints, identifying generalization operations and their calling sequences, defining iteration degree of each operation as well as parameters of individual algorithms involved in each operation. Following paragraphs reveal the importance of structure recognition in generalization involving different dimension.

2.2.1 Structure recognition and 2D generalization

According to the conceptual framework by (Brassel 1988), the overall process of map generalization, as stated earlier, is thought of consisting of five steps: Structure recognition*, process recognition, process modeling, process execution, and data display*. The generalization constraints (scale reduction factor, map purpose, data quality, etc.) provide control for these tasks. The purpose of structure recognition is essentially to determine the relevant structures and the relative importance of the map elements of the input map, and identify the generalization operators that should be used to solve the given generalization problem. It was during 1980s when the importance of structure recognition was gradually realized during line generalization. The model by (Brassel 1988) was extended by McMaster and Shea (McMaster 1992), decomposes the generalization process into three operational areas: a consideration of the philosophical objectives *why* to generalize, a cartometric evaluation of *when* to generalize, and the selection of appropriate spatial and attribute transformations which provide techniques on *how* to generalize. The area on when to generalize is equivalent in scope to the structure and process recognition in the model of (Brassel 1988). In the area of how to generalize, a list of various operators is proposed as described in first chapter.

In 1997, 2D generalization took a great leap ahead when structure recognition was extensively applied by (Mackaness 1997) using a new technology called “agent”. Here the significant groups are explicitly represented and their interconnected generalization and the parts are managed. The agent approach uses a multi-level, pyramidal model of agents. Single agents (like buildings) are called *micro agents*. On top of these micro agents, *meso agents* are designed, that correspond to groups of spatially organized features: a set of aligned buildings, a building block (set of buildings surrounded by roads), a town, etc (figure 12). *Micro agents* perform individual generalization without considering their surroundings: they apply generalization algorithms to themselves, like dilation or simplification, in order to satisfy their internal constraints. It means the recognition of internal structural of a graphic object is essential for micro agents. The constraints that involve several agents are designed and handled at the level of *meso agents*, e.g. a building block agent is in charge of handling the overlap conflicts between its roads and buildings. Therefore, structure knowledge for the objects in neighborhood is also needed and has to be known in advance for the meso agents.

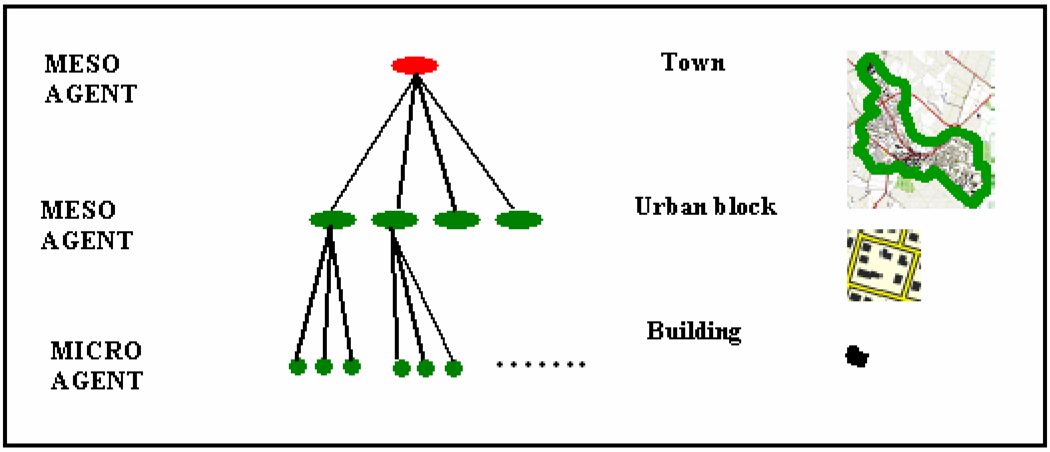


Figure 12: Pyramidal structure of AGENT (Duchene 2003)

The agent system incorporates a strategy, which aims to resolve conflicts not by describing in great detail how certain conflicts should be resolved, but by describing the desired final characteristics, i.e. structure recognition, of the feature. For example, the system might incorporate the desired outcome that no building should be smaller than a certain size, that it should not be closer than a certain distance to another building and that it must not be moved by more than a certain distance from its original starting position. The outcome of any generalization operation or set of operations is compared against this set of guidelines. This comparison is used to decide whether further operations are required, or whether the result should be discarded and a different operation applied. In this way, individual map features can be generalized in a way, which is sensitive to their particular situation; with similar features potentially having quite different operations applied to them.

(Duchene 2003) proposed modified approach which goes beyond the limits of pyramidal organizations listed above, by proposing another organization of agents to handle the relational constraints. *Relational constraints* are applied to a relation between two geographic objects, e.g. the constraint that prevents symbols overlaps. This approach is suitable for low-density areas such as rural areas, where the pyramidal approach does not work well. Here the structure recognition of graphics objects in neighborhood relations is more important.

As in the approach of the agent, agents here also need capacities of introspection in order to assess and satisfy their internal cartographic constraints. The major difference is that the consideration of the new capacities is also needed by agents in order to tackle their relational constraints. A relational constraint of an agent always involves the agent itself, and another agent that is within its neighborhood. This is why, in order to assess their relational constraints, agents need capacities of perception of their environment, as well as an explicit representation of this environment.

Moreover, to decide how to act in order to solve a relational conflict, an agent needs to gather information about itself (its state, what it can do or not), but also about the other agent involved in the conflict. In some cases, it is not able to compute all the information itself and needs to get it from the other agent. This is why new agents need to communicate with their neighbors as illustrated in Figure 13. The road agent which is a dead-end (yellow) can compute its symbol overlap with the building agent (black). The building agent can compute its proximity with the red road. In order to undertake the right action, the dead-end agent needs to know that the building is stuck. Otherwise, it should wait for the building to move away, because it is a cartographic rule that, *a priori*, moving a building has fewer consequences than moving a road. The information can be transferred from the building to the dead end thanks to a dialog such as "Move away! No, I cannot". This is why the agents need to have conversations, i.e. "task-oriented, shared sequences of messages that they observe, in order to accomplish specific tasks"

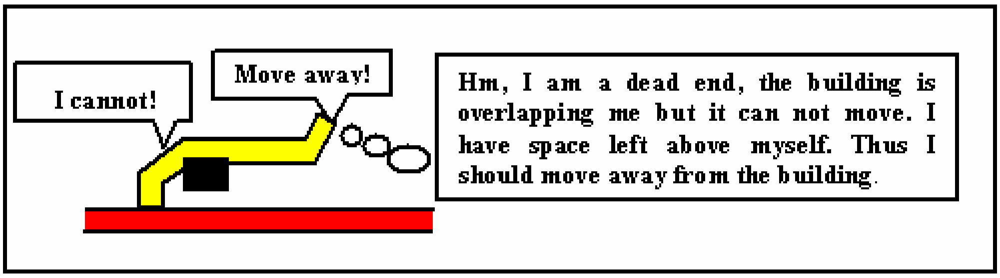


Figure 13: AGENT communication [Vincent 2003]

Three kinds of relational constraints are considered:

- Non-overlap constraint. No overlap should occur between (1) roads and buildings, (2) buildings and buildings.
- Proximity constraint. The distance between roads and buildings, and buildings and buildings must respect a given separability threshold (parameter of the system, 0.1mm on the map in our case).
- Topology constraint. A building is not allowed to "jump over a road".

Moreover, an internal constraint of positional accuracy for the buildings that prevents a building from moving too far away from its original position is considered.

Advantage of these innovations is that the representation of an agent's environment by means of constrained zones is of great help to choose a position that satisfies several relational constraints at a time. Moreover, it is very encouraging that the system and the system stemming from the agent project can work together. It must be noted here that whatever is the approach used, agents need to know structure recognition of the objects on which they are acting upon.

2.2.2 Structure recognition and line simplification

Cartographic generalization can largely be considered as a classification problem, since both structure recognition and process recognition are indeed classification tasks. It was during 1980s, when line generalization algorithms considered the importance of structure recognition. For example, during line (road) simplification, its structure recognition was required so that its different parts can be simplified or even exaggerated (accident prone bends).

ANN are good at *classification and b*y understanding generalization as a special type of classification problem, it should be possible to exploit the specific strength of ANNs in classification therefore (Weibel 1994), applied ANN for line generalization. They adopted the criteria for the ANN input representation used by (Mokhtarian 1992) which specifies that a general-purpose shape representation method should satisfy the following criteria, although these criteria were formulated for use in computational vision (e.g., for the recognition of objects):

The major reason why these requirements should be met is to avoid an overload of the network. Otherwise, two lines with the same shape, but linearly transformed, would not be recognized as identical and stored twice. The computation of form properties is important because the network only then can extract parts of learned lines. It shows that structure of the linear features are implicitly recognized which governs the subsequent generalization using neural network.

2.2.3 Structure recognition and displacement

(Monika 2000) applied Least Squares Adjustment theory (Jodio 1990) for the displacement of Map symbols**.** Displacement is also modeled as an optimization procedure, where the position of objects has to be optimized with respect to some given constraints. Different objects have to be displayed on a map – for reasons of legibility certain constraints have to be satisfied, e.g. minimal object sizes and minimal object distances have to be enforced. Least Squares Adjustment offers a straightforward framework to introduce different kinds of these constraints. In one step, all these constraints are solved simultaneously, resulting in one optimized solution with the feature that all residuals are distributed

Constraints can be two-fold. On the one hand, there are the exterior constraints in terms of minimal distances between objects that have to be enforced. On the other hand, there are the internal constraints of the objects, namely form parameters of the objects. The following set of constraints is introduced in the system.

- *Form parameters*: object sides, angles, orientation,
- *Distances between objects*: minimal distance that has to be enforced and critical distance, that indicates that the objects have to be merged (by setting the distance to zero)
- *Additional parameters*: the coordinates.

These constrains are derived only after a successful structure recognition of the map symbols which yield these form parameters and proximity.

All these observations R are introduced into the conventional LSA. They form the Jacobean Matrix **A**. As these functions are not linear, they have to be linearised with respect to given approximate values. Each observation has a corresponding accuracy (or weight), described in the matrix P. These weights describe how well the observation has to be enforced. They can be used to describe different object properties: the objects can be movable or fix, or they can be deformable or stiff. The stiffness is ensured by assigning a high weight to the internal form parameters of the object (object sides and angles). The introduction of the unknowns as additional parameters allows the assignment of accuracies. This can be used if an object is considered non-movable by assigning high weights to its coordinates.

After all the observation equations are set up, the solution of the unknown parameters X is gained by solving the following equation:

$$\mathbf{X} = (\mathbf{A}^T\mathbf{P}\mathbf{A})^{-1}\,\mathbf{A}^T\mathbf{P}(\mathbf{R} - \mathbf{f}\,(\mathbf{X}_o))$$

Where **A** is the Jacobean Matrix of the derivations of the functions according to the unknowns, **P** is the weight matrix, **R** are the observations and $\mathbf{f(X_o)}$ is the value of the function calculated at the approximate value $\mathbf{X_o}$.

As distance constraints between all the objects are formulated, this ensures that a global solution is found, where a displacement of one object occurs in accordance with all its surrounding objects.

2.2.4 Structure recognition and buildings generalization

Automatic generalization of 3D buildings based upon scale space has been tried by (Mayer 1998) where the formally well-defined theory of scale-spaces is introduced and structure recognition is the prerequisite. More specifically, an approach is proposed by him, which simplifies two dimensional (2D) as well as 3D building outlines employing vector-based morphology and discrete/continuous curvature space. The proposed approach renders it possible to preserve and even enforce right angles.

In continuation of above work, (Forberg 2003) proposed an approach for the simplification of 3D building data, which extends the idea of scale spaces applied in image analysis. Two scale spaces, a 3D version of mathematical morphology and the so-called 3D curvature space are applied separately, as both are suited for the simplification of different object structures. However, a rather complex analysis is needed for curvature space, which may dramatically slow down the generalization process. In a new approach introduced in (Forberg 2004a), the advantages of mathematical morphology and curvature space have been united in one process. Here, if the distance between two neighboring parallel facets falls below a predefined threshold, one or both facets will be moved towards each other until they merge into the same plane (Fig. 14). Such a "parallel shift" may lead to the simplification of all parallel structures including the split or merge of different object parts, the elimination or adjustment of local protrusions, step as well as box structures.

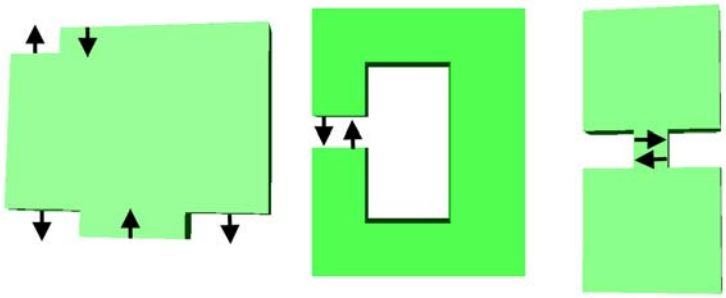


Figure 14: Parallel facets under a certain distance are shifted towards each other, until the facets of the building merge. (Forberg 2003)

For the results shown in Fig. 15, two kinds of weighted movements are used. If the area of a facet is smaller than a third of the area of its partner facet (this threshold is chosen intuitively), it will be shifted the whole distance. Otherwise, both facets will be shifted half of the distance. In this way, a shape simplification and adjustment take place simultaneously, which may slightly emphasize certain structures. The selection of a facet pair and the specific shift distances are based on the analysis of the relations between the facets.

If there are several pairs of parallel facets with the same smallest distance, one pair is selected randomly for the parallel shift. Therefore, the result is not always predictable. In case of small box structures, this problem can be avoided by shifting only the smaller facet the whole distance. Since the result remains the same now in spite of random choices, the symmetry of box structures can be preserved. However, the weighting does not work for all cases. The 4th building from the left of Fig. 15 shows a special case for this problem, where one facet has the same distance to two parallel facets. The random selection of the partner facet has led to an unexpected result, the elimination of one of two nearly symmetrical building parts. A human would most likely to close the gap instead.

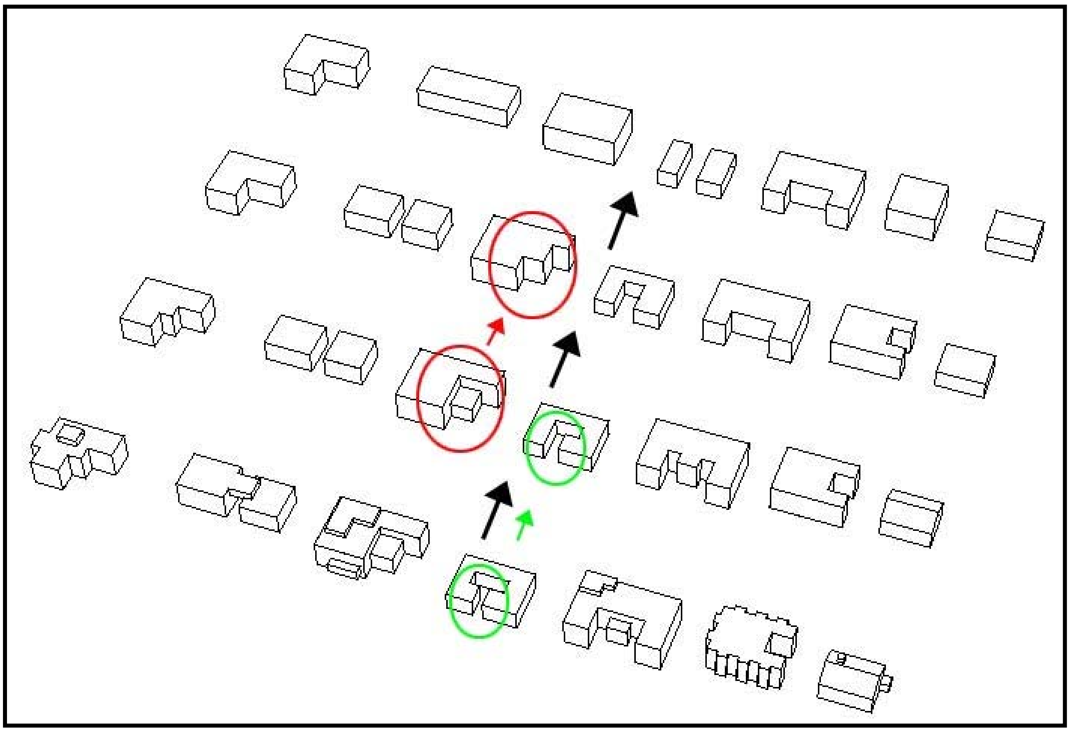


Figure 15: **Results for the simplification based on parallel shifts. Object parts marked in red are not only eliminated, but adjusted as well, so that the characteristic shape is preserved and slightly emphasized. Objects marked in green reveal the problem caused by random selection of one facet for the movement (Forberg 2003)**

The simplification using the parallel shift works only on strictly parallel structures. In order to simplify object parts with non-orthogonal structures, particularly roofs, a method for enforcing right angles is developed in (Forberg 2004b), according to which the inclined roof-facets are forced to become horizontal or vertical by rotating around one of their edges, i.e. the eaves or ridges. Among these two kinds of edges (taper-edge), which edge will be chosen for the rotation process and whether a horizontal or a vertical flattening (taper-orientation) occurs depend on the combination of two neighboring facets, the facet sharing its eaves and the facet sharing its ridges.

Recently a concept was introduced by (Thiemann 2002) for the generalization of individual 3D building which rely on surface simplification based upon edge, vertex and face reduction so that there is not appreciable difference between the original and the generalized building. The building is modeled as a combination of the main body and the features. These features are combined using Booleans operators especially *union* and *difference* and in some cases *interaction*. It presents a step-by-step procedure consisting of a segmentation of the building parts of a CGS representation, which are subsequently generalized. Apart from simplification by smoothing and omitting other generalization are also used such as emphasizing, aggregation, classification and displacement. As these operators require information about the importance of the objects and therefore have to be extracted from geometry (size, form) and topology (i.e. structure recognition). Non-spatial attributes are also used.

Different models are generated based upon LODs in order to achieve visual continuous representation and only incremental changes are stored to minimize the storage as shown in figure 16.

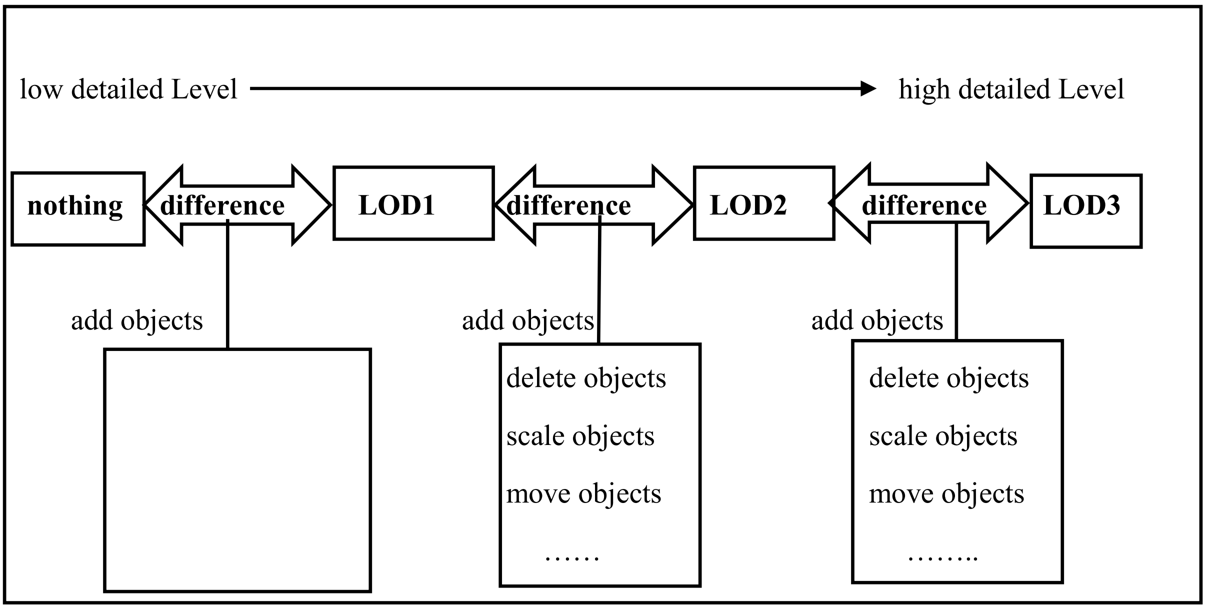


Figure 16: Differential multi scale data structure: storing differences between adjacent levels of detail

The comprehensive literature survey on structure recognition and its application in cartographic generalization has clearly shown the importance of structure recognition for generalization. Although structure recognition has been implicitly applied for all 2D generalization concepts and 3D building simplification but it is difficult to find enough literature, where 3D structure recognition and generalization are studied in details. The next chapter will concentrate on structure description of different 3D buildings where various measurable parameters are identified based upon various relations among these buildings.

Chapter 3

Structure description of 3D buildings

3.1 Structure description

Structure description of 3D objects is the first step towards its structure recognition and it includes information about the global shape of an object, as well as the relationships among the object's part. As objects can be recognized by their structural descriptions, therefore it is necessary to study structure description of 3D buildings. A 3D city model consists mainly of buildings and roads. There are a great number of building types in the world but the focus of the study in this thesis is on European buildings only. These buildings exhibit enormous variation in their geometric structural and shape. A different set of parameters is needed to describe each of them. Main emphasis is to maintain topology with explicitly described spatial relations. The geometric model is a straightforward description method. Various kinds of geometric models are described in the literature and each model required different parameters to describe 3D objects. Following are the most commonly used models.

3.1.1 Geometric model of buildings

The geometry of the buildings in urban area in particular, varies from simple to complex in shape and structure, from small to large in size, connected with the internal representation (i.e. deep structure) and external structure (i.e. surface structure) of the buildings. Individual 3D buildings regardless of their complexity can be described in one of the following ways (Lang 1999), (Thiemann 2004).

a. Voxel model: A 3D building is organized as a matrix composed of voxels. Each voxel can be attached to one or more semantic attributes. The voxel model shares the same advantages and disadvantages with 2D raster data. On the one hand, it can model the arbitrarily complex 3D buildings and allows direct access as well as simple image processing operations. On the other hand, it requires large storage capacity and long rendering time even if data compression methods such as octree have been applied.

b. Parametric model: A 3D building is partitioned into a number of standard bodies such as cuboid, sphere, cylinder, cone, and pyramid, which can be completely described by a few parameters such as side length, width, radius and height. The absolute position of the building is defined by six further parameters of rotation and translation. Parametric description is suitable for the description of simple buildings, which are characterized by their planar roof surfaces and the orthogonal relationship between walls and ground plans.

c. Constructive solid geometry (CSG): The solid geometry of a 3D building is constructed through Boolean operations such as intersection, difference, union or inversion of elementary building parts. The sequence of operations is stored in the CSG tree. Usually the length of the sequence reflects the relative complexity or irregularity of the corresponding 3D building.

d. Solid representation (Srep): The geometry of an arbitrarily complex 3D building is described by a Tetrahedral Network (TEN) composed of the regular or irregular topological elements tetrahedron, triangle, edge and vertex (*Song*, *Liu* and *Niu* 2004). Analogue to TIN, a TEN is usually based on the principle of Delaunay Tetrahedral Tessellation (DTT). Also, constrained DTT can be realized in which characteristic points, structure lines and planar facets of the building serve as vertices, edges and meshes of tetrahedrons. The most popular algorithm for constrained DTT is the "Boundary Face Subdivide (BFS)" algorithm. It uses the initial input vertices to form an initial TEN, then inserts constraining vertices into the mesh until all the constraining lines and facets are recovered.

e. *Polyhedral models*: These general models can be used to represent all types of buildings bordered by flat surfaces(Forberg 2002). Usually they are represented by means of their surrounding boundaries. This so called boundary representation (b-rep) consists of information about each of the faces, edges, co-edges and vertices of the building, and how they are connected as shown in figure 17. This is a suitable representation for graphics since the surfaces to be visualized are readily available. Based on this model, buildings with different structures and roof styles can be drawn. The properties of these individual entities are described in table 1.

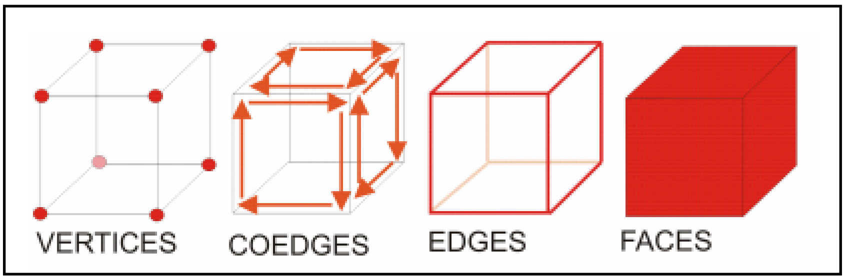


Figure 17: Important elements of the building(Forberg 2002)

Building parts	Description
VERTEX	Vertices are described by 3D coordinates and are connected by edges.
EDGE	In a manifold solid object, every edge is related to exactly two faces and two co-edges.
CO-EDGE	A co-edge is a directed edge. The two co-edges related to an edge always point in opposite directions along the edge and are associated with a loop of one of the faces.
LOOP	A loop is a connected series of co-edges and describes a boundary of a face. Generally, loops are closed, having no start or end point.
FACE	A face is a bounded portion of a surface.
BODY	Body is the highest level of model object and in our case describes the entire object that is composed of faces.

Table 1: Building parts as per b-rep model (Forberg 2002)

3.1.2 Semantic model of buildings

The structure of polyhedral buildings is shown in the geometry layer of the semantic network (figure 18). Polyhedrons, e.g., buildings, consist of their surrounding regions called *faces*. The boundary of a region consists of lines (edges, co-edges), which connect points (vertices).

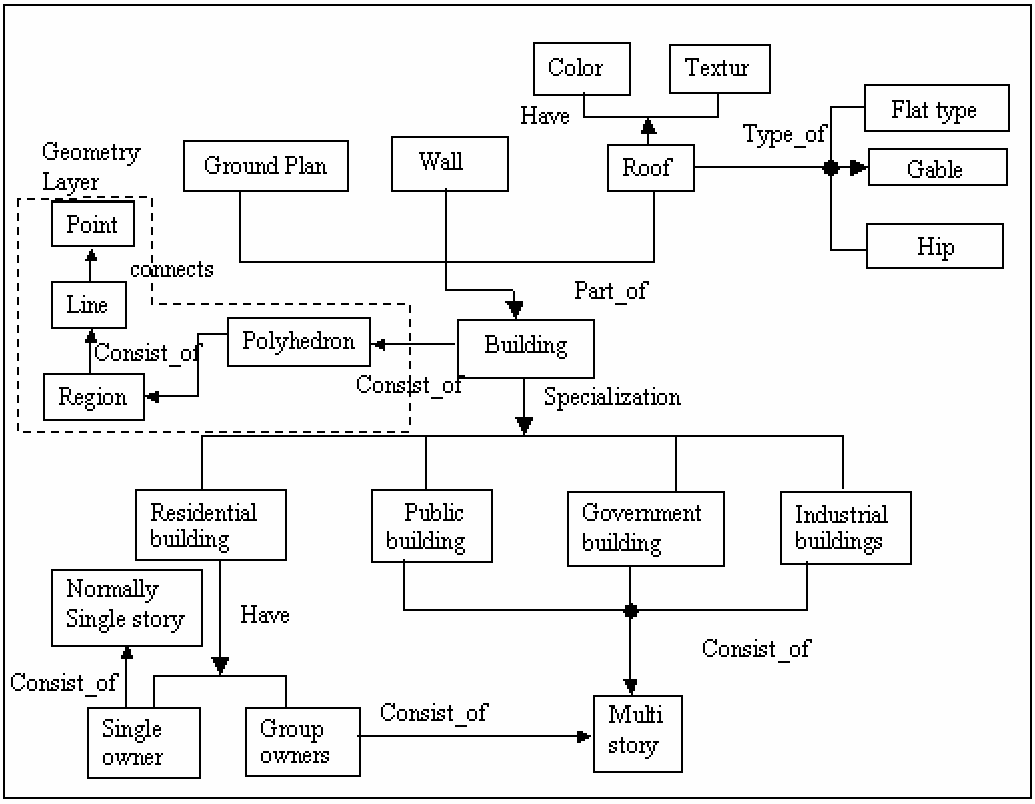


Figure 18: Semantic model of buildings

3.2 Semantic modeling of a 3D city

When modeling a 3D city the main characterizing structure of the individual objects as well as the semantic relations between them has to be preserved. A semantic network on the conceptual level (Brachmann 1997) is developed and is shown in figure 19. A typical European city consists of different kinds of buildings and houses and after the World War II, the variety of these buildings has increased to many folds due to industrialization. A visible renovation of the old town, the erection of new and transformation of old industrial sites into commercial buildings has taken place. In the city's social structure, the most evident changes are the ongoing gentrification of the old town and the new kind of sharp segregation arisen with the new single family housing areas. Therefore, a city area can be segregated into various categories such as residential, commercial, industrial, and agricultural depending upon many factors. According to the density, the residential area is divided into small houses and highly constructed buildings comprising of many apartments. Residential house density and their height give information about the population density and are in directly proportional. Height of the house also indicates income level of the residents. In case of higher income, the house would be higher and bigger and have good and different shapes. These houses are relatively away from the city center and owned by mostly native peoples. They also located little away from the main road and connected to it by small road.

On the other hand, high-storied buildings consist of many apartments and highly populated. On an average, people are not rich and have middle level income. These people are not native but have come from other parts for doing job. These buildings are relatively near the center of the city. They are located near to the maid roads around it. It has also a relatively big garden near it and possibly a small river passing nearby. Commercial areas are always in the central part of the city and well approached by roads from all directions. It is again very densely populated. The buildings are multistoried and of the same shape and height. Industrial buildings are located outskirt of the city and approached by small roads linked to main roads. They are aloof and surrounded by open field area

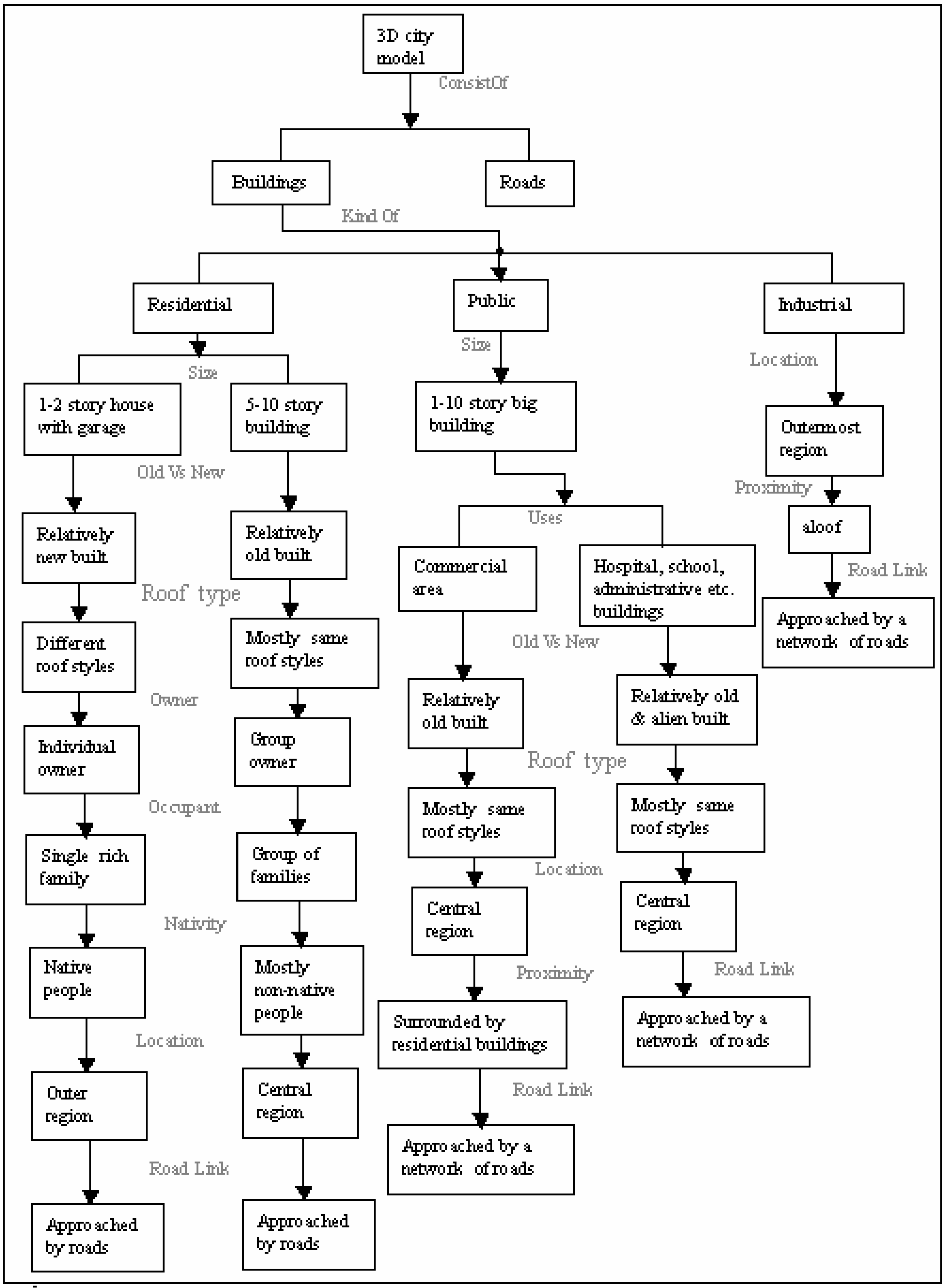


Figure 19: Semantic network of settlement (Brachmann 1997)

3.3 Building types based upon shapes and roof styles

Buildings, which are the major constituent of the city, are manmade objects and reveal a high variability in structure. A building can be represented as a combination of several simple building parts such as roof, walls, ground plan, windows etc. Each part in turn is represented by sets of vertices, edges and faces. When there are no attributes attached to them, they correspond to point, line and region respectively. A *point* is defined uniquely by its coordinates in a coordinate system on $\mathbf{R}^2$. An *arc* is a straight-line segment with two distinct points, a *start point* and an *end point*. A *line* is a finite sequence of simple arcs a_1, a_2...a_n such that the end point of the arc a_i coincides with the start point of the arc a_i+1, for every 1≤ *i*≤ n-1. The endpoints of a line are called *nodes*. If nodes of a line coincide, it is called *closed*, otherwise it is called *open*. A *simple closed line* has a non-crossing interior. A *region* is a connected subset of 2D space which is characterized by a simple closed line called the *outer boundary*, denoted by l_0 and potentially a set of simple closed lines called the *inner boundaries*, denoted by l_i for i>0, such that

- No two such lines intersect.
- Neither of them is contained in the internal set of the other and
- All of them are contained in the internal set of the outer boundary.

The region then can be expressed by the formula (Vasilis 1997)

$$\cap\ i>0\ (ex(l_i)\cap in(l_o))$$

Geometrically and Semantically these building parts can be divided into three levels of abstraction (Lang 1999)

- **Feature level.** Feature level contains features, namely attributed vertices, edges and faces as shown in figure 20. Attributes for edges and faces, for instance, are the orientation classifications such as horizontal, oblique and vertical. Faces have an additional attribute describing its role: valid values among others are wall, roof and floor. These features can be represented spatially as well as non-spatially. In general, the set of spatial parameters are divided into positional parameters on one hand, describing location and orientation, and form parameters on the other hand like length, width, and height.

Figure 20: Feature level

- **Feature aggregate level.** Feature aggregate level contains feature aggregates, which are induced by vertices, edges and faces, and contain all their direct neighbors. Each aggregate is defined by a feature graph, given by a set of features and adjacency relations. A corner, for instance, contains one vertex and all its adjacent edges and faces (figure 21).

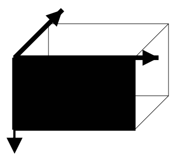


Figure 21: A corner containing vertex, edge and face

- **Building level.** This level contains complete building. A typical city model consists of buildings of different roof styles, shape, sizes, colors, textures etc. The most prominent among them is roof style and therefore these buildings can be classified into a known set of groups based upon their different roof styles:

3.3.1 Types of roof

The following major roof types are identified in a typical European city model:

a. Flat roof: Flat roof consists of only one single plane. In term of faces, it has four faces (walls) joined together at right angles and one face (roof) lying at the top upon them (see figure 22.a).

b. Gable roof: Gable roof is composed of two intersecting planes that form a peak (the ridge) between the planes (figure 22b). In other words, roof consists of two rectangular faces joined together at an acute angle and two triangular faces inserted between the gapes of two (see figure 20.b).

c. Salt box roof: Salt box roof consists of two faces - the small front face and the large rear face. Front face makes an angle of approximately 45^0 and rear face makes an angle of 20^0 with the ground wall (see figure 22.c).

d. Cross gable roof: Cross gable roof consists of pairs of gable roofs set at right angle to each other. The description of roof faces is the same as that of gable roof (see figure 22.d).

e. Gambrel roof: Gambrel roof is similar to a gable roof. However, rather than having a single ridge at the peak, a gambrel roof has three ridges, one at the peak and the two along the sloping sides. Therefore there are four faces joined together making acute angles with each other and two other arc-shaped faces inserted perpendicularly between the four faces (see figure 22.e).

f. Hip roof: Hip roof is similar to a gable roof but has four surfaces instead of two. It is made of four intersecting planes. Two rectangular faces are joined together at acute angle and two triangle shaped faces are inserted perpendicularly on either side of them (see figure 22.f).

g. Dutch hip roof: Dutch hip roof is a combination of a hip and a gable roof with its central section made of gable roof (see figure 22.g).

h. Shed roof: Shed roof is one that starts at the eaves of the existing roof and continues at a lower pitch. It has only one slanted face emerging from the existing roof and leaning on the wall faces (see figure 22.h).

i. A-frame roof: A-frame roof uses a steep roof to form the walls of the upper level. It consists of two rectangular faces, which form the roof as well as walls in A-shape. Two triangular shaped faces, forming front and rear part of the buildings are inserted perpendicularly to rectangular faces (see figure 22.i).

j. Mansard roof: A mansard roof has two slopes on each of the four sides. The lower slope is steeper than the upper slope. The upper slope is usually not visible from the ground. There are eight faces making up the roof. Four lower faces are acutely joined to form the lower portion of the roof and other four faces making upper portion of the roof like hip roof (see figure 22.j).

k. Pyramid roof: A pyramid roof is built on a square base with eaves of the same length. It is made of four intersecting planes. In another words, roof planes consist of four triangular faces joined together such that they meet at a single point (see figure 22.k).

l. Hip with cross gable roof: Hip with cross gable roof is a mixture of hip and gable roof with hip roof at the center and gable crossing it (see figure 22.l).

m. Lean to wall roof: Lean to wall roof has one sloped roof face with upper end leaning against the wall face (see figure 22.m).

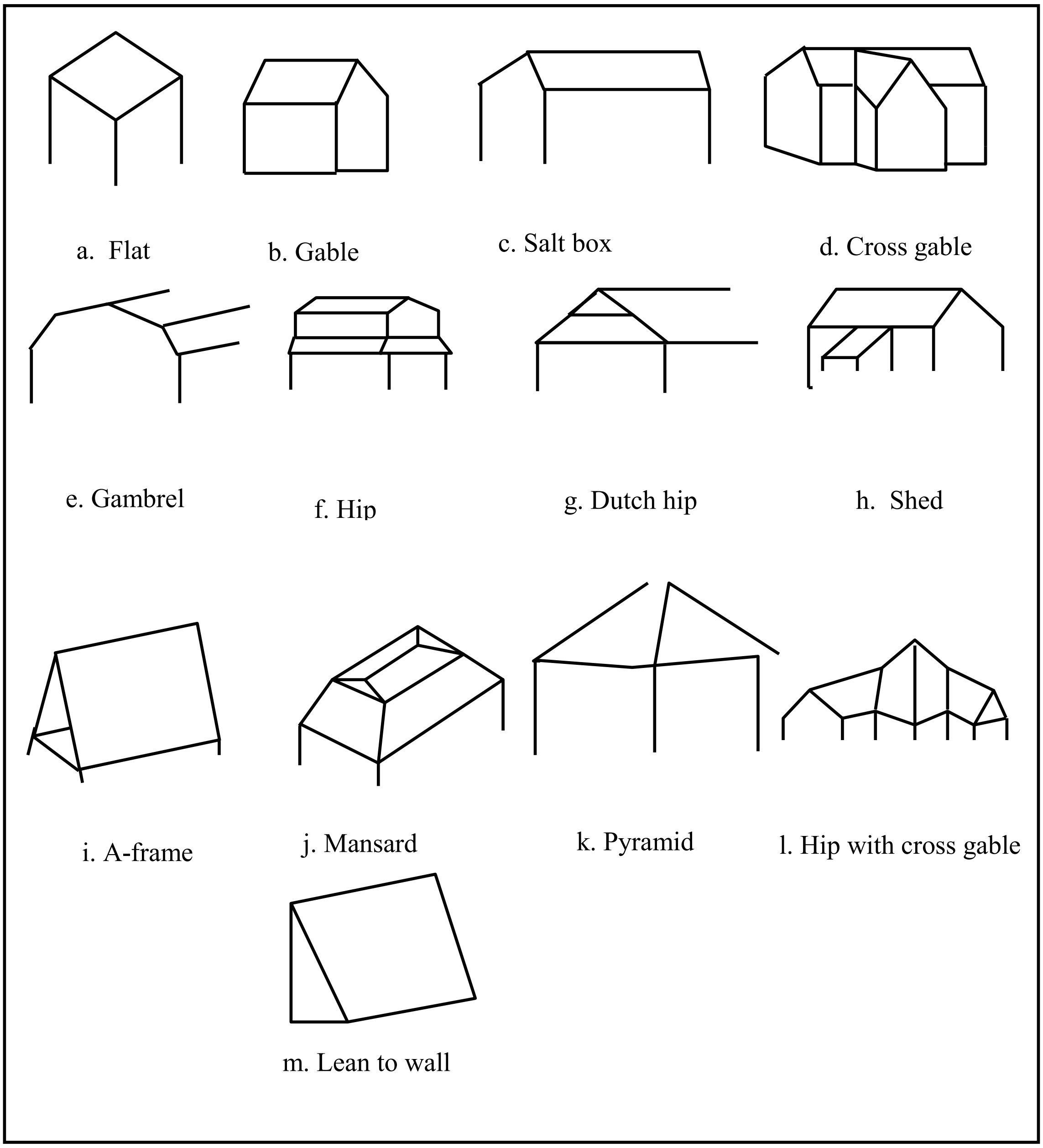


Figure 22 : Different roof types

3.3.2 Building types

One of the most important criteria for defining classes of buildings is its shape complexity. Here, for a city model, buildings can also be classified based upon different types of roof and shape complexity.

1. **Simple buildings:** These usually corresponding to individual housing with different roof style and ground plan (figure 23). It has normally rectangular ground plan and orthogonal walls. Even it doesn't have rectangular ground plan, it can still be broken into different rectangles. Roof is also made from planer faces combined together to form a given style. As these buildings are occupied by a single family, so mostly they are 1-2 story buildings.

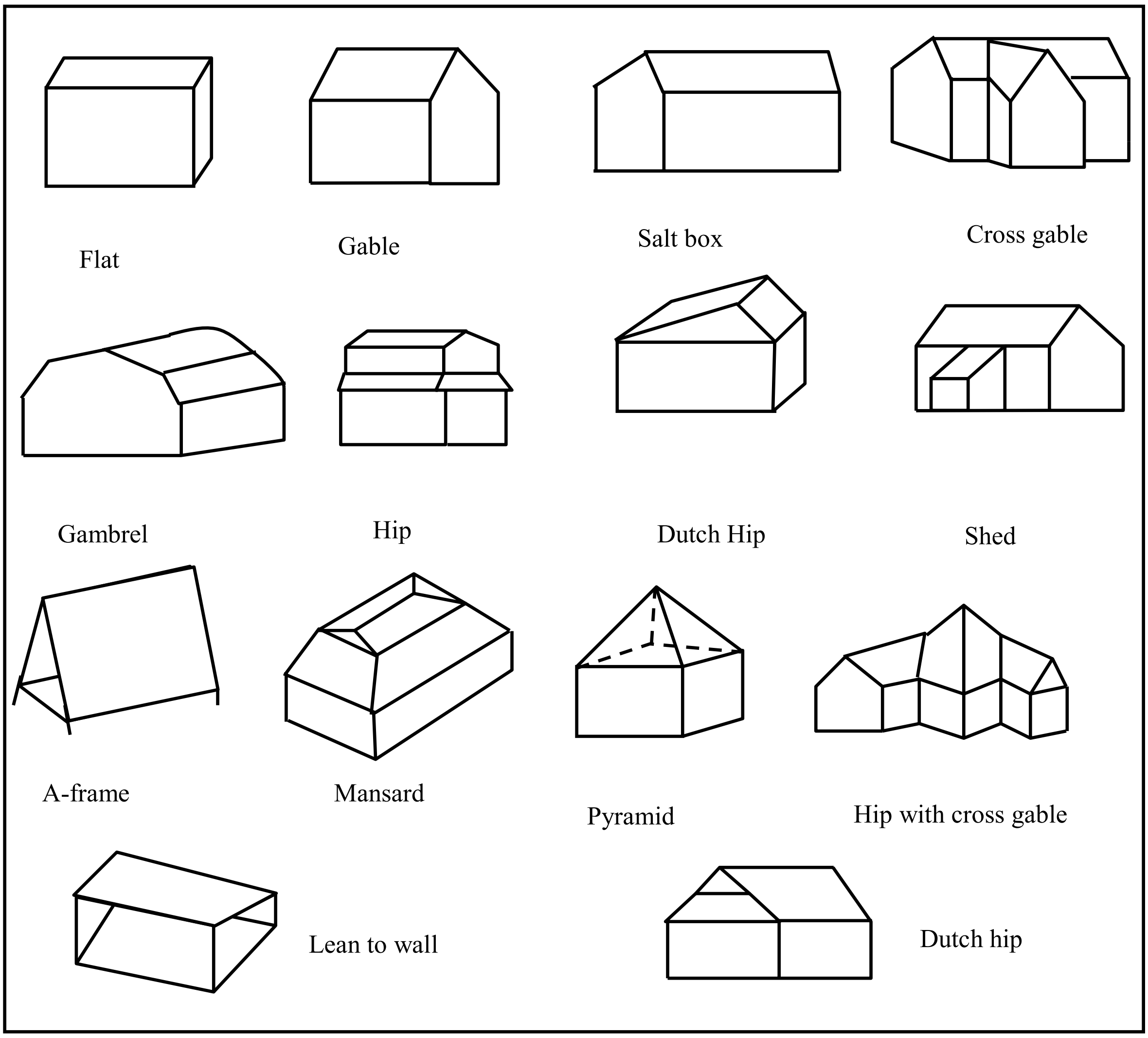


Figure 23: Simple building types

2. **Complex buildings**: They are amalgamation of buildings in a town centre. These Buildings are formed from a group of small buildings joined together (figure 24). For example, U-Type, L-type or Trapezium building is formed due to clustering of many adjacent buildings. Two buildings are joined together in the middle through a common path, thus forming a bridge building.

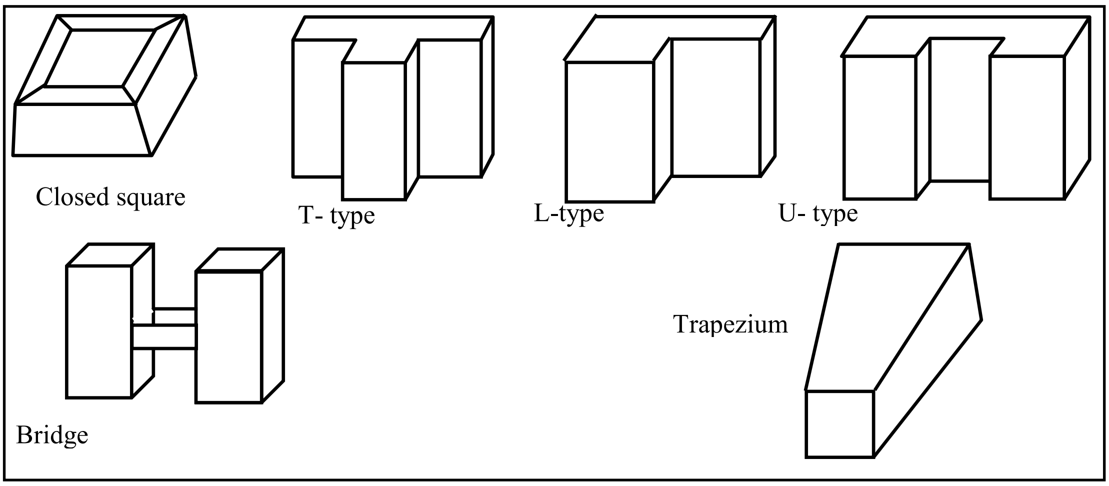


Figure 24: Complex building types

3.4 Building parameters

A set of measurable parameters is required to characterize the size and shape of a building. The minimum number of parameters varies from building to building. According to (Suveg 2001), following are the details of parameters for each type of buildings.

i. Flat type: The minimum set of parameters required for flat type building are its side length l and width of the ground plan w, height of the body h, x,y,z coordinates of building reference point and the number of faces. (figure 25).

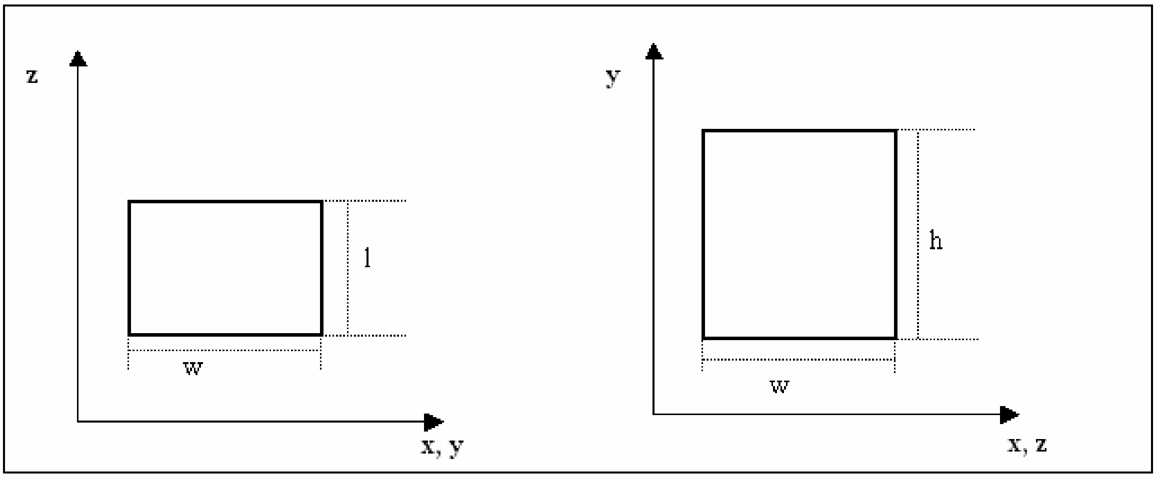


Figure 25: Some of the parameters of flat building

ii. Gable roof: The minimum set of parameters required for gable building are its side length l and width of the ground plan w, different heights (h1,h2) of the body, x,y,z coordinates of building reference point, the height of ridge, the number of faces, angle between the faces α and the roof type (figure 26).

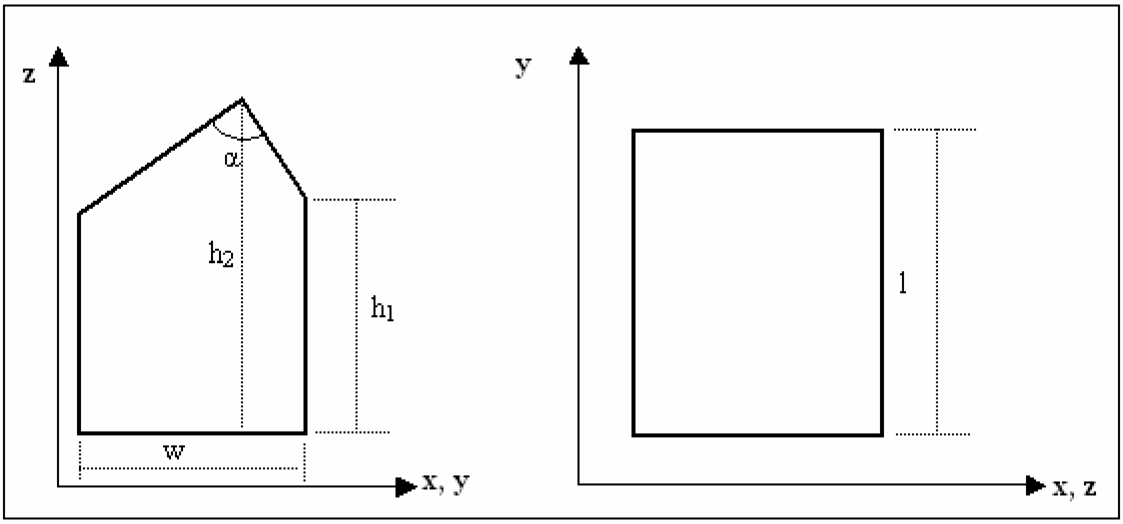


Figure 26: Some of the parameters of gable building

iii. Hip building: The minimum set of parameters required for hip type building are its side length l and width of the ground plan w, height of the wall h_2, height of the ridge h_1, x,y,z coordinates of building reference point, the number of faces, angle between the faces α and the roof type (figure 27).

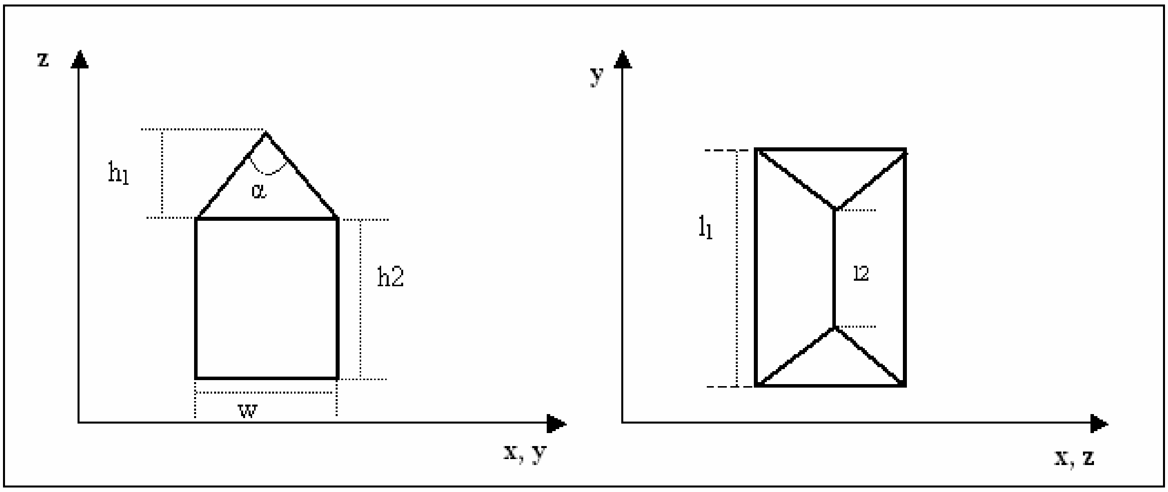


Figure 27: Some of the parameters of hip building

iv. Mansard: The minimum set of parameters required for mansard type building are its side lengths (l_1,l_2) and width of the ground plan w, three different heights (h_1,h_2,h_3) of the body, x,y,z coordinates of building reference point, the number of faces, angles between the faces α and the roof type (figure 28).

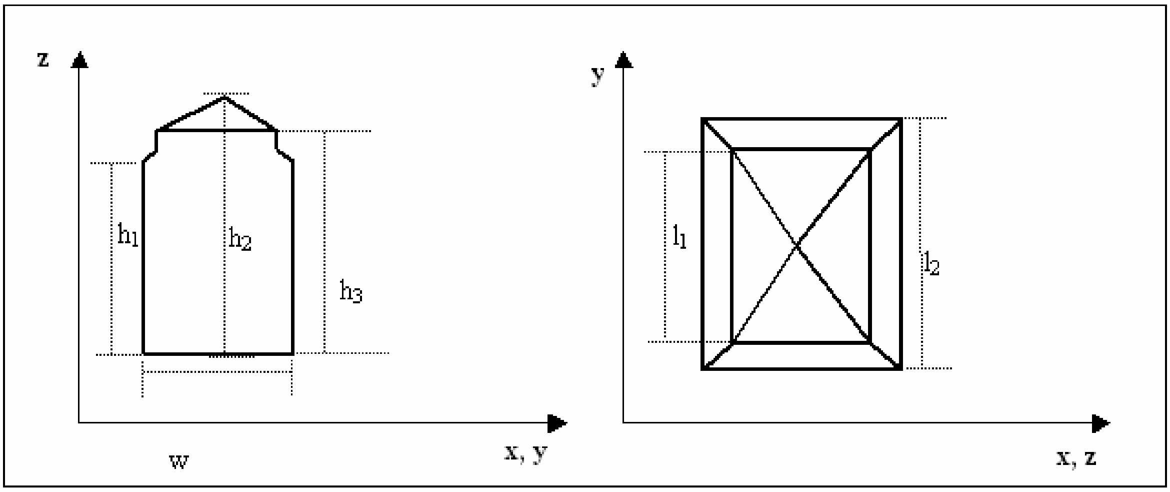


Figure 28: Some of the parameters of mansard building

v. Lean-to-wall building: The minimum set of parameters required for lean-to-wall type building are its side length l and width of the ground plan w, two different heights (h_1,h_2) of the body, x,y,z coordinates of building reference point, the number of faces, angle between the faces and the roof type (figure 29).

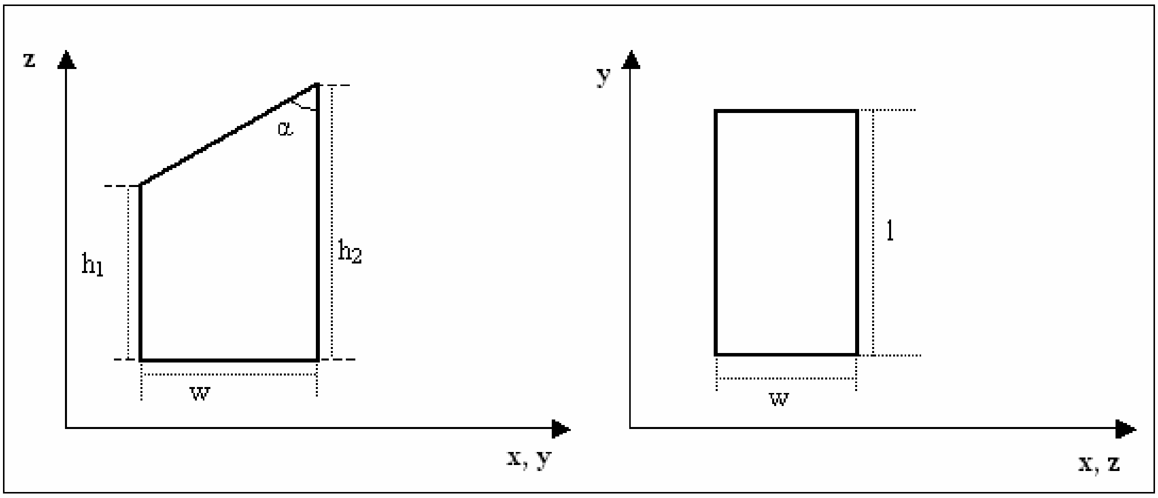


Figure 29: Some of the parameters of lean-to-wall buildings

vi. Pyramid: The minimum set of parameters required for pyramid type building are its side length and width of the ground plan, two different heights (h_1,h_2) of the body, x,y,z coordinates of building reference point, the number of faces, angle between the faces and the roof type (figure 30).

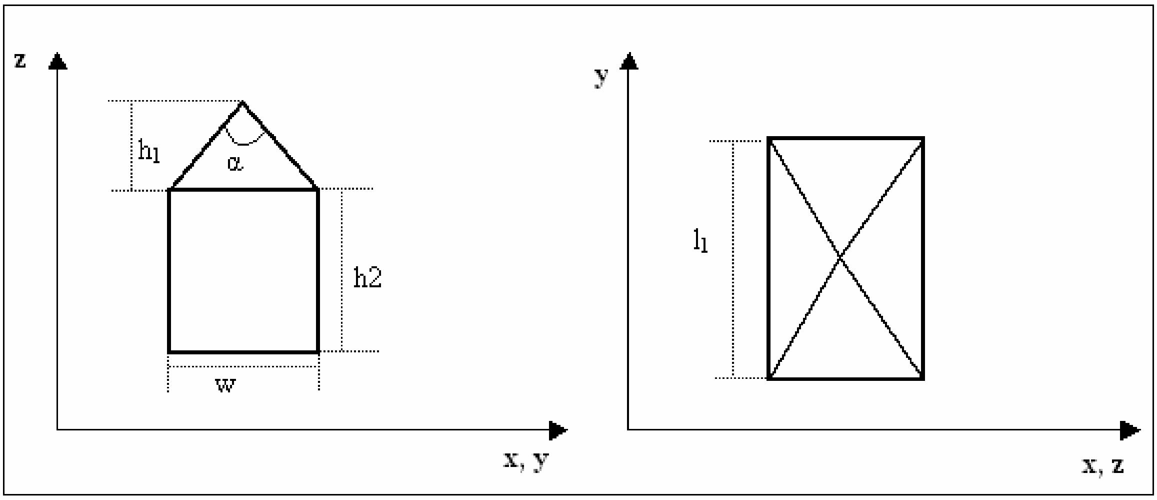


Figure 30: Some of the parameters of pyramid buildings

vii. A-Frame: It is similar to gable roof building but the roof sides touch the ground to make shape similar to letter A. The minimum set of parameters required for A-Frame building are its side length l and width of the ground plan w, height of the body h, x,y,z coordinates of building reference point, the number of faces, angle between the faces and the roof type (figure 31).

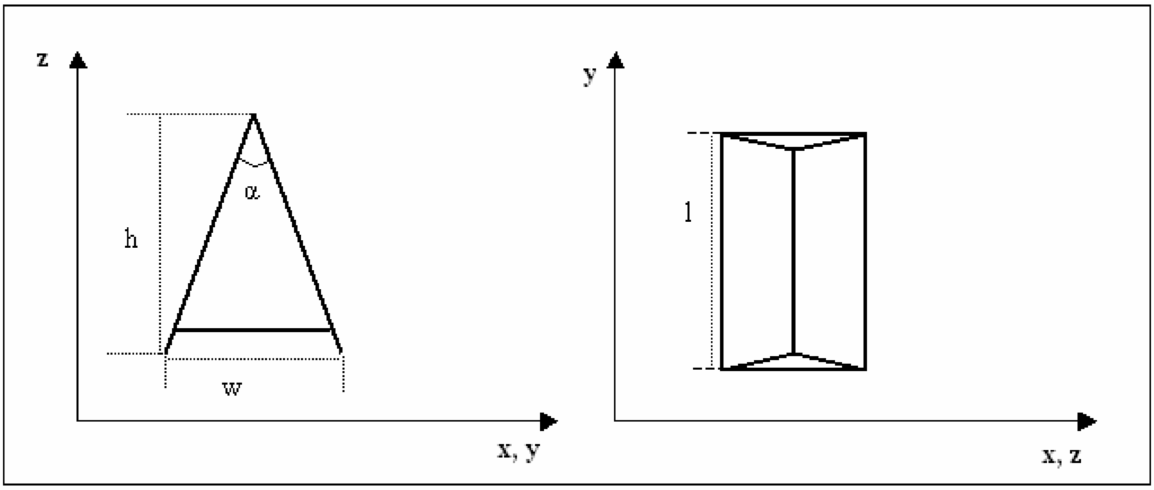


Figure 31: Some of the parameters of A-frame building

viii. Gambrel: The minimum set of parameters required for Mansard building are its various lengths (l_1,l_2,l_3) and width of the ground plan w, three different heights (h_1,h_2,h_3) of the body, x,y,z coordinates of building reference point, the number of faces, angle between the faces and the roof type (figure 32).

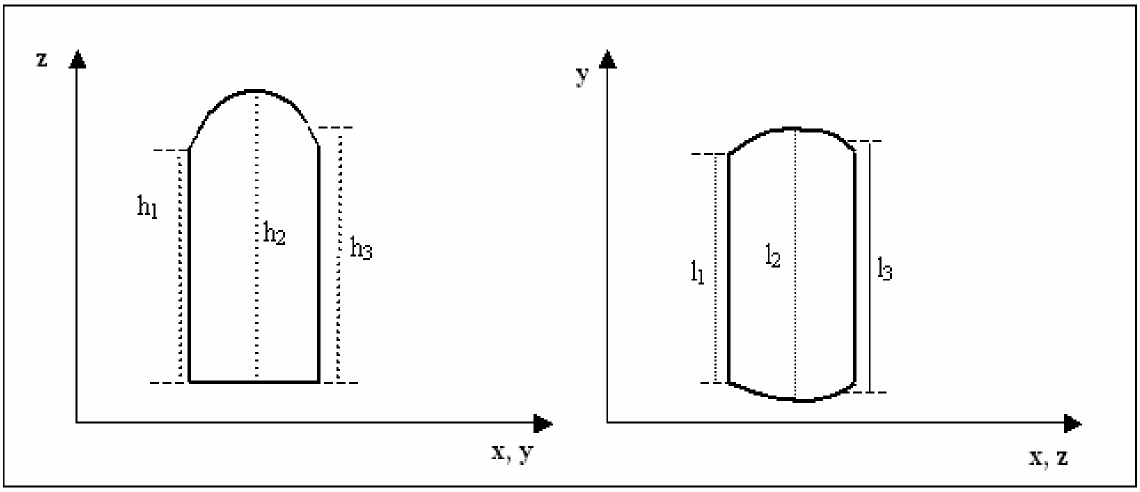


Figure 32: Some of the parameters of gambrel building

ix. **L,U,T building**: The minimum set of parameters required for L,U,T buildings are various side lengths (l_1,l_2) and widths of the ground plan (w_1,w_2), height of the body h_1, x,y,z coordinates of building reference point, the number of faces and the roof type (figure 33).

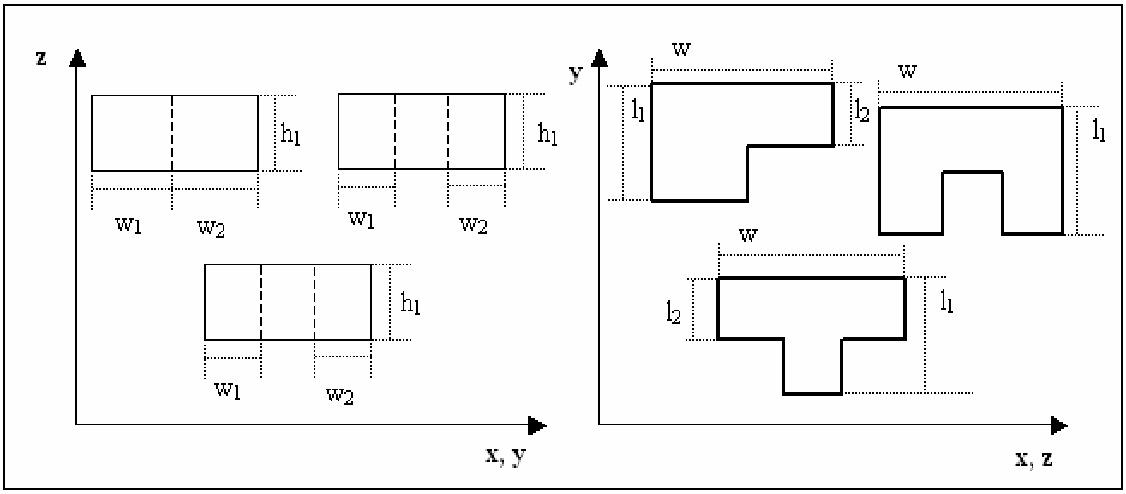


Figure 33: Some of the parameters of (L,U,T buildings)

3.5 Hierarchical approach to structure description

Structure description of objects in a city model can be divided into three levels namely micro, meso and macro level based upon structure recognition at individual building, buildings in neighborhood and buildings at cluster level. The following paragraphs study these levels in details.

a. Micro level: It applies to individual objects (buildings). A building exhibit a large number of significant edges, junctions, parallel lines, faces and closed figures composed of polyhedrons (Kada 2002). The higher-level features of the buildings exhibit apparent regularity and relationship, and play important role in structure recognition. An individual 3D building can be

a. Roof type
b. General shape
c. Positional parameters describing position and orientation
d. Form parameters like length, width, height, surface area, volume
e. Orthogonal walls (in most cases)
f. co terminations
g. L and U junctions
h. Parallelism of faces
i. Shape symmetry
j. Shape regularity
k. Straight line segments

characterized by

Measurable parameters for buildings are:

1) **Junctions**: L and U type junctions are formed when different faces of the building are attached side by side and are approximately at right angle to each other.

2) **Parallel edges and faces**: Most of the buildings have faces and edges (i.e. walls), other than the roof plane, are parallel and orthogonal to each others.

3) **Shape regularity**: Shape regularity is defined in term of its edge linearity and shape convexity. Regularity of a building is measured by its convexity measure defined by the ratio of the area of the shape to the area of its convex hull (Christophe 2002).

R_S = Area(Shape)/ Area (Convex hull)

Where $\mathbf{R_S}$ is the measure of shape regularity and always varies from 0 to 1.

4) **Shape symmetry**: Most buildings have very strong symmetry where the left side of the structure is exactly the same as the right side. Although it is typically viewed as a discrete feature where an object is either symmetric or non-symmetric, however visual perception treat symmetry as a continuous feature, relating to statements such as `one object is *more* symmetric than another' or ‘an object is *more* mirror symmetric than rotational symmetric’. With this notion in mind, it can be viewed as a continuous feature and a Continuous Symmetry Measure (CSM) is defined to quantify the "amount" of symmetry of different shapes and the `amount' of different symmetries of a single shape. Computational methods have been developed (Zabrodsky 1992) to compute the CSM values of a shape P:

$$CSM(P) = d(P, ST(P))$$

Where ST is the *Symmetric Transform* defined as the symmetric shape closed to P in term of the metric d.

5) **Length, width and height**: These are the basic important measurable parameters of a building.

With these positional and form parameters, we are able to recognize an object from different viewpoints. The minimum values, where the different parts of the object are distinguishable are known as minimum dimensions. For a 2D map, the minimum dimensions mentioned in (Hake 2002) are for paper maps whereas corresponding values for 3D are for a computer screen resolution. Therefore, these values are little more than its counterpart in 2D for the obvious reasons. These minimum dimensions of the single 3D object at a given scale are given in the table 2.

Parameter	Minimum (or Threshold) size
Length, width	$l_{min} =$0.40mm, $w_{min} =$ 0.40mm
Height	$h_{min} =$ 0.45mm
Angle	$\alpha_{min} = 5^o$
Area	$A_{min} = l_{min} * w_{min}$
Volume	$V_{min} = l_{min} * w_{min} * h_{min}$

Table 2: Minimum dimensions

6) **Orthogonality**: Though not all, most of the buildings have orthogonal boundary walls. Orthogonality of the walls can be measured by calculating the normal to each wall and measuring

the angle of the various junctions formed between the various walls of the building. Normally the value measured should be between 80^{o} to 100^{o} as shown in figure 34.

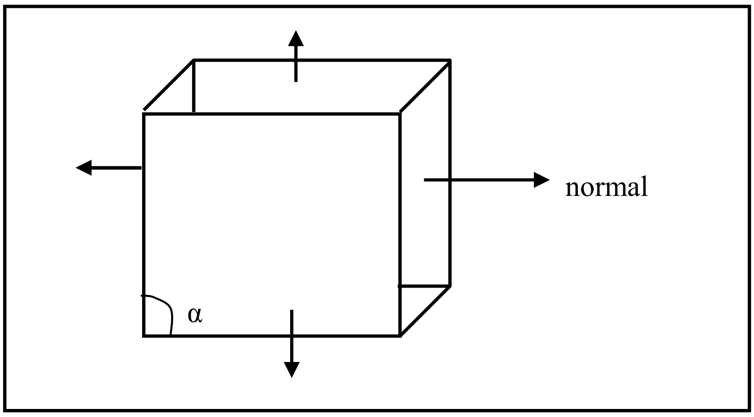


Figure 34: Orthogonality of a building

7) **Area & volume**: Building area and volume are other easily measurable parameters. While area can be calculated using building length and width, volume uses its height also. However, when the building is not a perfect cube, calculating volume using length, width and height gives incorrect result. Therefore finding the convex hull of the building and then calculating its volume gives better result.

8) **Shape deviation**: Though most of the building shapes may match to one of the shapes described above, but a building may not match exactly into one of the above categories and their deviation has to be measured. Therefore a new index, called shape index, Γ, is defined to measure this deviation, using ground plan of the building, as follow:

$$\Gamma = 1 - [\text{area (E and S)} / \text{area (E or S)} + N_E/N_S\ ]$$

Where E is the original shape from above category and S is the shape to be compared as shown in figure 35. N_E is the number of rings (loops) of E and N_S is the number of rings of S shape

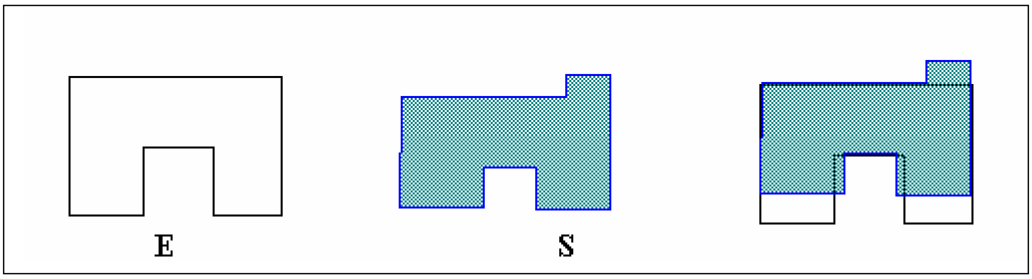


Figure 35: Shape deviation

Most of the parameters (viz. orthogonality, area, volume etc.) above discussed can be calculated using ACIS Geometric modeler functions (www.spatial.com) and remaining parameters functions have been implemented. Table 3 gives the list of the functions, from ACIS and self implemented, used to calculate these parameters:

Sr. no	Function name	Description
i.	api_get_vertices	Gets all vertices related to an entity
ii.	api_get_faces	Gets all faces related to an entity
iii.	api_get_loops	Gets all loops related to an entity
iv.	api_get_edges	Gets all the edges related to an entity
v.	api_get_lumps	Gets all lumps related to an entity
vi.	api_get_tco-edges	Gets all the co-edges related to an entity
vii.	GetBldgRoofType();	Gets roof style
viii.	GetBuildingArea();	Gets area of the building
ix.	IsBodyOrtho();	Returns if the building is orthogonal
x.	GetBodyColor();	Gets the color of the building
xi.	GetBodyType();	Gets the type of the building
xii.	api_point_in_body	Check if a given point is within the building
xiii.	api_entity_extrema	Gets the extreme range of the building
xiv.	Parallel	Check if the walls are parallel
xv.	GetRoofHeightNExtent	Gets the roof height, wall height and extent of the building
xvi.	PtToPtDist	Distance between two points in a building
xvii.	api_get_entity_box	Gets the bounding box
xviii.	Get_owner_transf	Gets general 3D affine transformation of building

Table 3: Minimum dimensions

b. Meso level: It applies to an object in relation to its neighbors. The following set of spatial relationships among the neighbors can be identified:

a. Proximity
b. Relative orientation
c. Nearest neighborhood
d. Roof style contrast
e. Shape contrast
f. Height difference
g. Aloof
h. Alignment
i. Size contrast

These spatial relations among the buildings are shown below in figure 36.

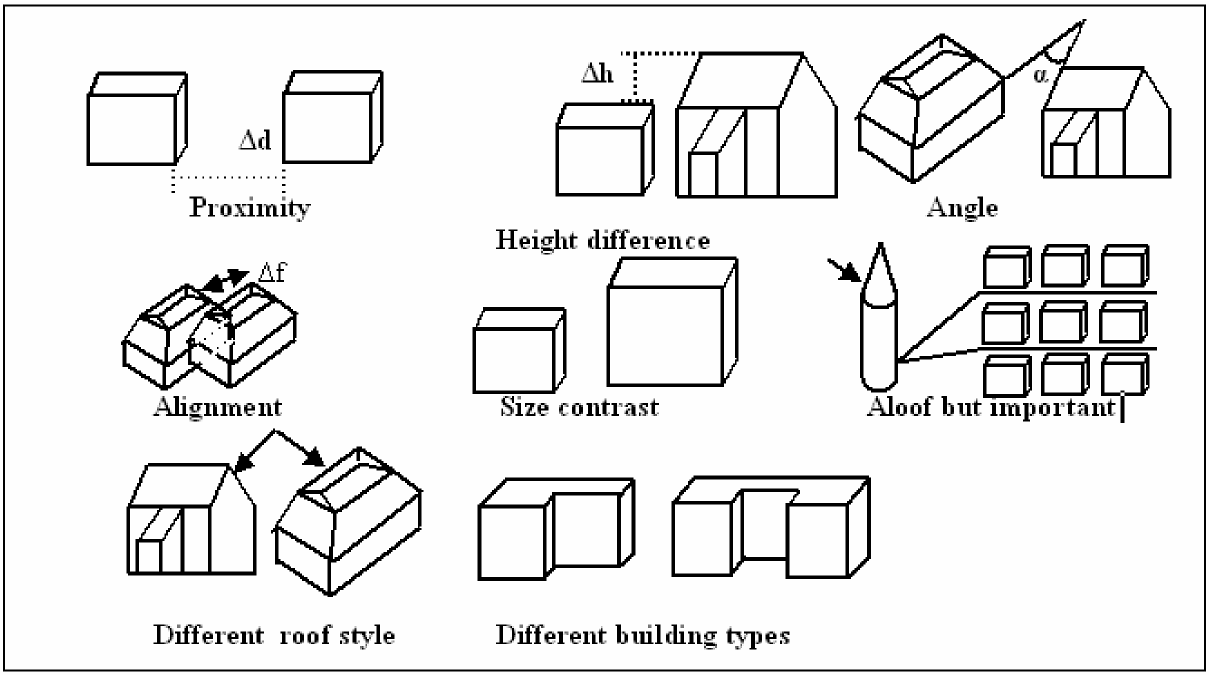


Figure 36: Neighborhood relations

Measurable parameters at this level are:

1) **Distance and proximity**: Distance and proximity are closely related terms. Both of them may give rise to conflict between two objects. Every distance related conflict is also a violation of the proximity constraints but *not* vice versa (Peter 2001). While a distance conflict requires being resolved (e.g. because two vertices of a object are too close to each other), proximity expresses rather an option; we can, for instance, aggregate two objects of the same category because they lie within a specified distance of each other. The shortest distance at which we still can clearly visually separate two objects or identify all parts of one object depends upon the chosen target scale and the viewpoint. Beside this, it is also influenced by various other *map controls*, for instance, screen resolution.

2) **Height difference**: Height difference is always measured from ground to the highest point of the two buildings irrespective of their different roof types i.e. plane or gable. Minimum height difference is also user-defined and scale dependent.

3) **Nearest Neighborhood (NN)**: Nearest neighborhood gives the number of nearest neighbors to a given object and distance between them. Two cases are:

   - restricted by extent of neighborhood by distance
   - restricted by extent of neighborhood by direction

   If there is no nearest neighbor of an object, it implies that it is an isolated entity and helps us to decide whether it should be deleted or exaggerated depending upon its importance.

4) **Relative orientation**: It gives the angular difference between two nearest faces of the neighboring buildings. The minimum angle, that makes two objects distinguishable, depends upon scale, screen resolution and the viewpoint of the user.

5) **Mutual alignment**: Mutual alignment of two objects is also a deciding factor in determining if they will be eligible for generalization.

6) **Aloof**: It means there is no nearest neighbor (i.e. within a given distance) to a given object as described above and is treated as aloof.

7) **Roof style contrast**: Two neighbor objects may have their roof of different styles.

8) **Building type contrast**: Different types of two neighbor buildings

Using Voronoi diagram and Delauney's triangulations method, we can find most of the proximity related parameters such as nearest neighbor, distance between them, angle between them and their alignment as shown in figure 37.

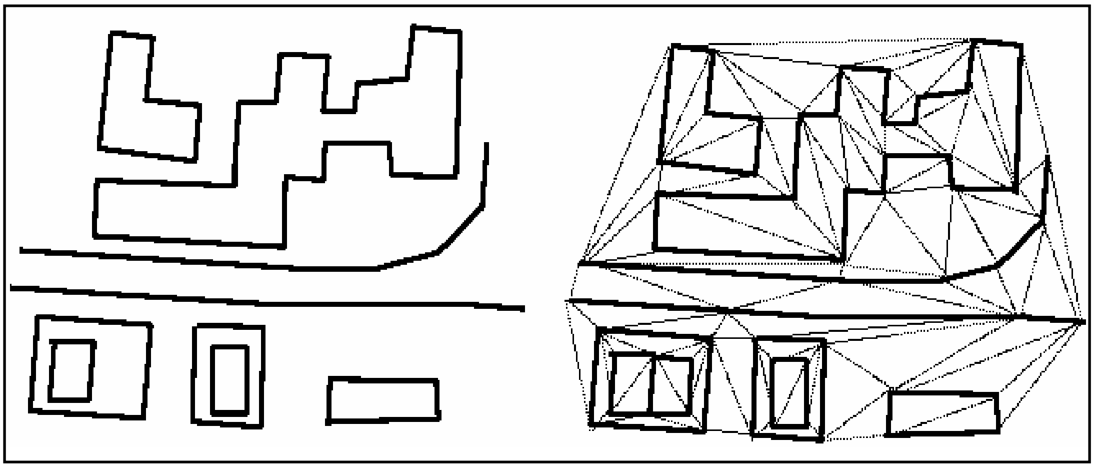


Figure 37: Proximity measurements using delauney triangulations (L. Meng, 2002)

Table 4 gives a set of implemented functions used to measure various proximity parameters

Sr. no	Function name	Used for
i.	Angle_between	Orientation
ii.	FindDistancefromEntity	Proximity, aloof
iii.	EntitiesHghtDiff	Height difference
iv.	GetBuildingArea	Building size
v.	GetLowerExtreme	Alignment
vi.	GetUpperExtreme	Alignment
vii.	GetBodyColor	Color of the body
viii.	BodySmall	Size of Building
ix.	BodyWideApart	Proximity
x.	api_entity_entity_distance	Distance between two buildings
xi.	D3GenDelaunay2D	Runs Delauney algorithm
xii.	GetTriangleQuantity	Returns the number of triangles
xiii.	GetTriangle	Returns a given triangle
xiv.	GetVertex	Gives the vertex of the given triangle

Table 4: Proximity functions

These minimum dimensions for above parameters related to proximity are given in the table 5.

Parameter	Minimum size
Distance	$\Delta d_{min} = 0.30$mm
Height difference	$\Delta h_{min} = 0.40$ mm

Angle	$\alpha_{min} = 5^o$
Mutual Aaignments	$\Delta f_{min} = 0.30$ mm

Table 5: Proximity parameters

c. Macro level: Macro level applies to clusters of objects having similar properties such as settlement blocks and is based on psychological observations of humans. Different parts of the city exhibit different cluster densities and these differences have to be preserved. Macro level

a. Shape, size, and regularity.
b. Height and vertical regularity.
c. Roof shape and color.
d. Adjacency to other structures.
e. Unique, deterministic features

structure recognition helps to maintain this. While doing so, the main emphasis is on:

Measurable parameters for an object are:

a. Building roof type,
b. Building shape
c. Building colors and texture
d. Building height
e. Building similarity
f. Building size

All these parameters of individual building are already known in the meso level and have been utilized in recognition of building structure forming a cluster in chapter 6. But before that, these buildings are recognized using ANN in chapter 5 and chapter 4 gives a brief introduction ANN before they are applied in chapter 5.

Chapter 4

Artificial neural networks and structure recognition

The lack of efficient automated generalization tools in GIS is due to the fact that generalization is a difficult task (Sébastien 1999). One of the reasons is that it is guided by a lot of geographic and cartographic knowledge. The approach of building expert systems has proved efficient in numerous fields where knowledge requires to be introduced. In recent past, lot of research work has been done in identification of 2D cartographic objects and generalization using various AI techniques. Being inspired, an attempt has been made here in recognition of 3D buildings using ANN. However, it is necessary here to give a brief introduction of the technique so that an understanding of the whole technique is made more easier and simple before it is being applied in next chapter.

4.1 Neural network overview

The brain is composed of a very large number (~ 10^{10}) of massively interconnected neurons (Haykin 1994). There is an average of several thousand interconnects per neuron, although it varies enormously. Each neuron is a specialized cell as shown in figure 38, which propagates electrochemical signals. A neuron has a branching input structure (the dendrites), a cell core, and a branching output structure (the axon). The axon of one cell is connected to the dendrites of another via synapses. When a neuron is activated, it fires an electrochemical signal along the axon. This signal crosses the synapses to other neurons, which may in turn fire. A neuron fires only if the total signal received at the cell core from the dendrites exceeds a certain level, the firing threshold.

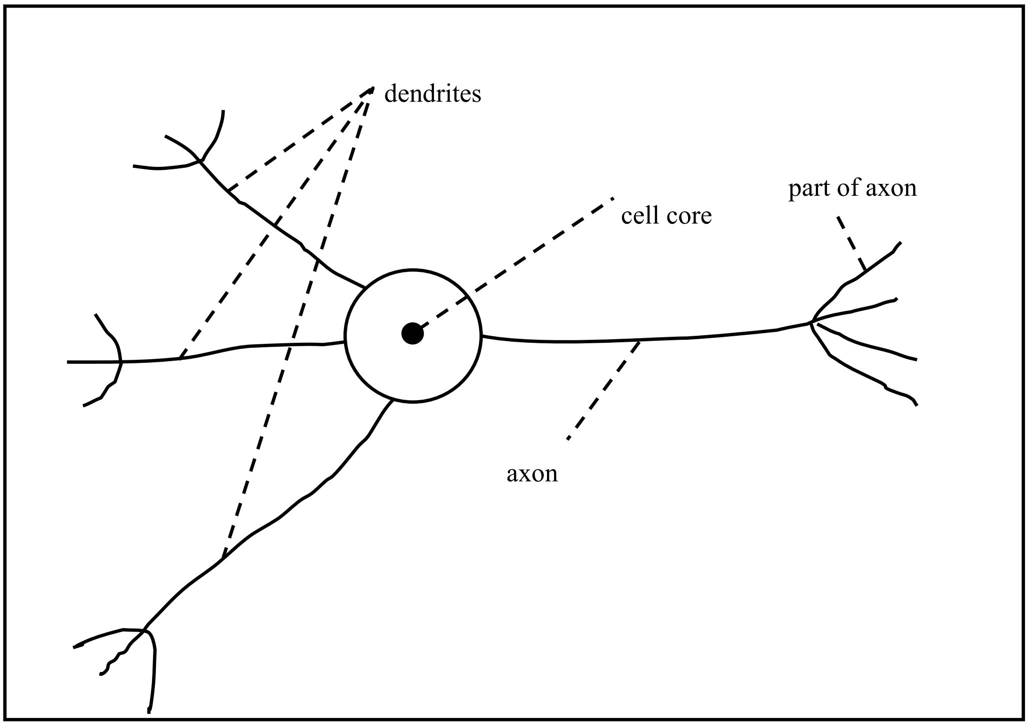


Figure 38: Parts of neuron

4.2 Artificial neural network

An artificial neural network (ANN) is an information-processing paradigm inspired by the way the densely interconnected, parallel structure of the mammalian brain processes information. There are many definitions of ANN given in literature:

According to (Haykin 1994), ANN is a massively parallel-distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. (Nigrin 1993) sees an ANN as a circuit composed of a very large number of simple processing elements that are neurally based. Each element operates only on local information. Furthermore, each element operates asynchronously, i.e. there is no overall system clock. From (Zurada 1992) point of view, ANNs are physical cellular systems, which can acquire, store and utilize experiential knowledge.

Intuitively ANN are collection of mathematical models that emulate some of the observed properties of biological nervous systems and draw on the analogies of adaptive biological learning. The key element of the ANN paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements that are analogous to neurons and tied together with weighted connections that are analogous to synapses.

To capture the essence of biological neural systems, an artificial *neuron* is defined as follows:

It receives a number of inputs, either from original data or from the output of other neurons in the neural network. Each input comes via a connection that has a *weight* as shown in figure 39. These weights correspond to synaptic efficacy in a biological neuron. Each neuron also has a single threshold value. The weighted sum of the inputs is formed and the threshold subtracted to compose the *activation* of the neuron, also known as the post-synaptic potential or PSP. The activation signal is passed through an activation function, also termed transfer function, to produce the output of the neuron

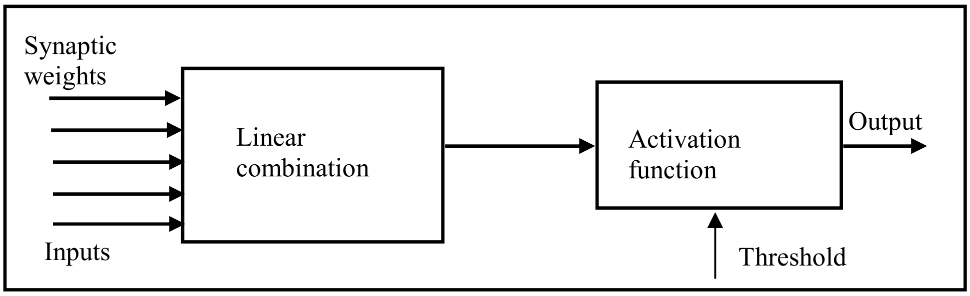


Figure 39: Parts of nerve cell

It can be seen from the above, that there is an analogy between biological (human) and artificial neural networks. The analogy is summarized in table 6.

However, it should be stressed that the analogy is not a strong one. Biological neurons and their neuronal activity are far more complex than might be suggested by studying artificial neurons. Real neurons do not simply sum the weighted inputs and the dendritic mechanisms in biological systems are much more elaborate. Also, real neurons do not stay *on* until the input changes and the outputs may encode information using complex pulse arrangements.

Sr.no	Biological neurons	Artificial neurons
1.	Neuron	Processing unit
2.	Dendrites	Combination function
3.	Cell core	Transfer function
4.	Axons	Element output
5.	Synapses	Weights

Table 6 :Biological Vs artificial neurons

ANN are able to detect similarities in inputs, even though a particular input may never have been seen previously. This interpolation capability also holds for noisy input data. ANN may be used as a direct substitute for autocorrelation, multivariable regression, linear regression, as well as trigonometric and other regression techniques.

When data is analyzed using an ANN, it is possible to detect important patterns that are not previously apparent to a non-expert. Thus, the ANN can act as an expert. To detect a pattern, the ANN must first be trained by processing a large number of input patterns and showing it what output results from each input pattern. Once trained, the ANN is able to recognize similarities when presented with a new input pattern, resulting in a detected pattern.

4.3 Classification ANN

A typical ANN has N inputs and one output (figure 40). The input layer is composed not of full neurons, but rather consists of the data values that constitute inputs to the next layer of neurons. The next layer is called a hidden layer. There may be several hidden layers. The final layer is the output layer.

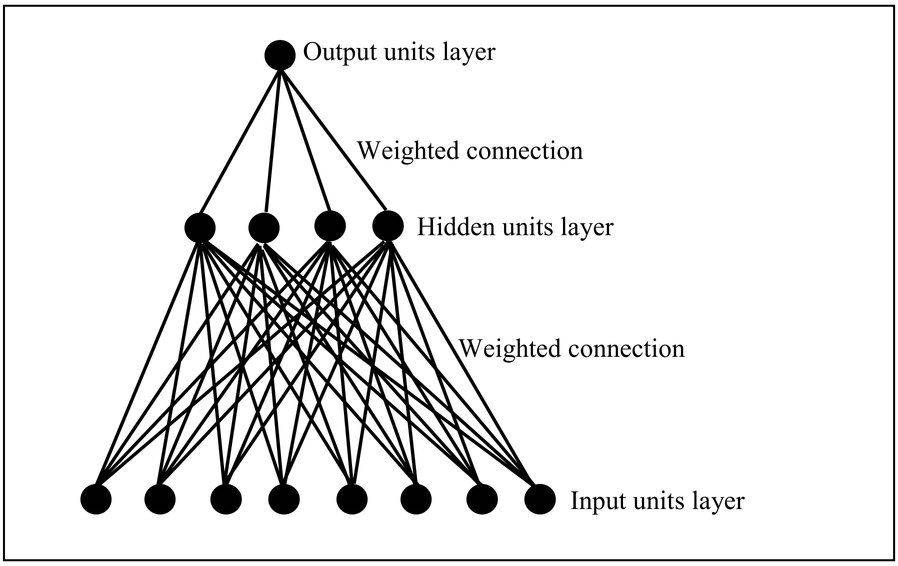


Figure 40: ANN structure

Using the *network architecture* as basis, there are three major types of neural networks:

- *Recurrent networks* - The units are usually laid out in a 2D array and are regularly connected. Typically, each unit sends its output to every other unit of the network and receives input from them. Recurrent networks are also called *feedback networks.* Such networks are "clamped" to some initial configuration by setting the activation values of each of the units. The network then goes through a stabilization process where the network units change their activation values and slowly evolve and converge toward a final configuration of "low energy". The final configuration of the network after stabilization constitutes the output or response of the network. This is the architecture of the Hopfield Model (Hopfield 1982).
- *Feed forward networks* – These networks distinguish three types of units: input units, hidden units, and output units. The activity of this type of network propagates forward from one layer to the next, starting from the input layer up to the output layer. Being sometimes called multilayered networks, feed forward networks are very popular because this is the inherent architecture of the Back Propagation (BP) Model.
- *Competitive networks* – These networks are characterized by lateral inhibitory connections between units within a layer such that the competition process between units causes the initially most active unit to be the only unit to remain active, while all the other units in the cluster will slowly be deactivated. This is referred to as "winner-takes-all" mechanism. Self-Organizing Maps (SOM), Adaptive Resonance Theory (ART), and Rumelhart & Zipser's competitive learning model are well-known examples for this type of network (Haykin 1994).

The network architecture can be further subdivided according to whether the network structure is *fixed*

- *Static architecture* – Most of the seminal work was based on static network structures, whose interconnectivity patterns are fixed *a priori*, although the connection weights themselves are still subject to training. *Perceptrons, multi-layered perceptrons, self-organizing maps,* and *Hopfield networks* all have static architecture.
- *Dynamic architecture* – Some neural networks do not constrain the network to a fixed structure but instead allow nodes and connections to be added and removed as needed during the learning process. Some examples are Grossberg's *Adaptive Resonance Theory* and Fritzke's *Neural Gas*. Adding-pruning approaches to multi-layered Perceptron networks have also been widely studied.

or not. There are two broad categories:

Yet another criteria for classifying ANN models is according to the modes of learning adapted. In this case, there are two major categories:

- Supervised learning – Supervised learning (Andina 2001) is performed under the supervision of an external teacher as shown in figure 41. Generally speaking, the teacher has the knowledge of the environment or the system. It guides the neural network in adjusting its parameters. The desired output pattern corresponding to an input is presented to the net during training in order to guide learning. The net learns in the training phase by adjusting its weights such that the actual net output becomes more similar to the desired net output. The perceptrons and back propagation networks are classic examples of supervised learning models.

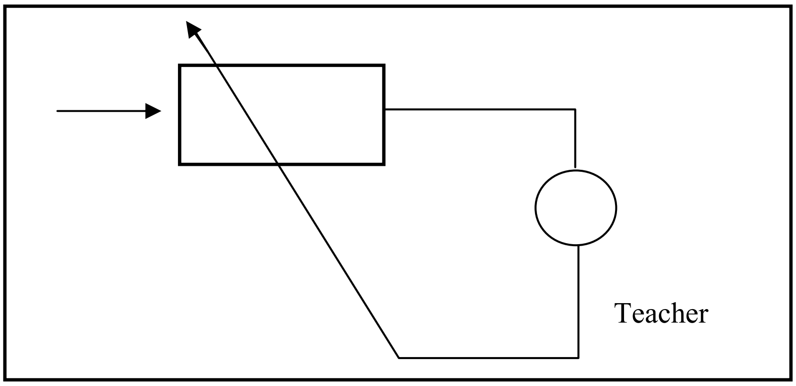


Figure 41: Supervised learning (Andina 2001)

- *Unsupervised learning* – When the desired output pattern is not available to guide learning or some ANN models do not need category information to accompany each training pattern, is called unsupervised learning (figure 42). But after learning, such information would still be required in the interpretation and labeling of the resultant networks. Classical examples of these are Kohonen's *self-organizing maps* and Grossberg's *Adaptive Resonance Theory*.

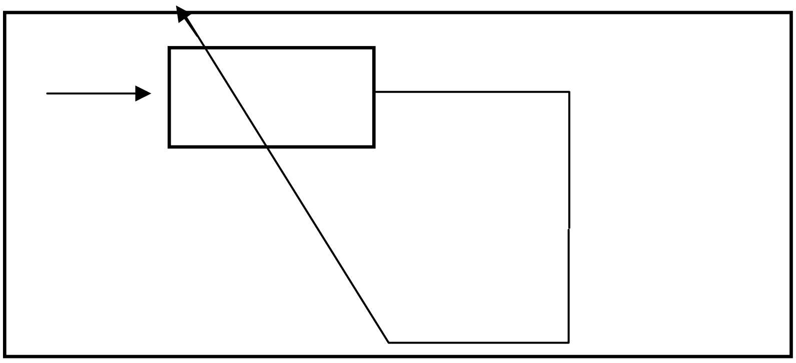


Figure 42 : Unsupervised learning (Andina 2001)

It also makes sense to classify neural network models (Seng 2000) on the basis of their *over-all task*:

- *Pattern association* – The neural network serves as an *associative memory* (Seng 2000)by retrieving an associated output pattern from some given input patterns. The association can be *auto-associative* or *hetero-associative*, depending on whether or not the input and output patterns belong to the same set of patterns.
- *Classification* – The network seeks to divide the set of training patterns into a pre-specified number of categories(Seng 2000). Binary output values are generally used for classification; although continuous-valued outputs coupled with a labeling procedure can do classification just as well. For binary output representation, each category is generally represented by a vector (sequence) of 0's; with a single 1 whose position in the vector denotes the category.
- *Function approximation* – The network is supposed to compute some mathematical function(Seng 2000). The network's output represents the approximated value of the function given the input pattern as parameters. In certain areas, *regression* may be the more natural term

There are other criteria for classifying neural network models, but these are less fundamental than those mentioned above. Some of these include the type of input patterns (binary, discrete valued, real valued), or the type of produced output values (binary, discrete-valued, real valued).

4.4 Characteristics of ANN

ANN have different architectures (Seng 2000) with learning schemes and varying weight update modalities, used to perform different tasks. Yet, some basic characteristics can be attributed to ANN:

- *Inherent parallelism* - In practically all ANN, many of their units work in parallel.
- *Parity of components* - To a very a large extent, the units of ANN have the same structure and behave similarly;
- *Access to local information* - ANN are composed of units that are interconnected with each other. Any given unit's level of activation and eventual output depends exclusively on its current state and the outputs of the other units to which it is connected. Whatever happens to other units in the system that are not connected to a given unit will not directly affect its actions.
- *Incremental learning* - ANN do not learn a given concept in one cycle. Instead, the network parameters undergo several small changes, and, over the time, they reach their final values.
- Over learning - ANN has the drawback of over learning. It could cause memorization where the network might simply memorize the data patterns and might fail to recognize other set of patterns. Thus, early stopping is recommended to ensure that the network learn accordingly.

This chapter gives a brief introduction into ANN but enough to understand the recognition of 3D structure using them. Following chapter devoted to 3D building structure recognition using ANN.

Chapter 5

Neural network approach to 3D building recognition

5.1 Knowledge abstraction

All the steps of 3D visualization not only involve a knowledge representation, but also include a knowledge abstraction. One of the important steps in 3D visualization is generalization which comprising of a knowledge representation process, when objects are symbolized, and a knowledge abstraction process, when objects relevant to the theory construction are identified. According to (Sébastien 1999), "Knowledge abstraction in generalization is the identification of abstracted geographic objects relevant to the theory construction that will be done from the map. "Objects" have to be taken here in a very wide sense: they may represent any basic geographic objects (like a house, a road…) or any set of basic objects having a geographical meaning (the set of streets of town, a street and the buildings along it…)". This chapter addresses the identification of various types of buildings, simple as well complex, using ANN.

5.2 Recognition of 3D buildings

A building is represented here using "B-rep". As described in previous chapters, this format is a layered description of geometric objects. The first layer contains vertices, the second layer one-dimensional edges, third layer consist of 2D faces and so on.

Using this representation, solids can be described unambiguously by their surface and topologically orienting it such that it is possible to tell, at each point of the surface, on which side of the surface the interior of the solid lies. This includes a topological description of the connectivity and orientation of vertices, edges, and faces, and a geometric description for embedding these surface elements in space. Further, the topological description specifies vertices, edges, and faces and indicates their incidences and adjacencies. The geometric description specifies, for example, the equations of the surfaces of which the faces are a subset. Most of the information available about a building using this representation will be used as the inputs to the ANN.

A 3D building consists of different parts, e.g. ground plan, walls, roof and building as a whole. Out of them, walls are mostly rectangular and have orthogonal faces and therefore not included in the recognition process. To recognize other features of the buildings, a hierarchical approach is followed. It involves recognition of:

- Ground plan
- Roof type
- Simple building
- Complex building

A city model of BONN is taken for recognition. It consists of approximately 400 different buildings ranging from simple to complex.

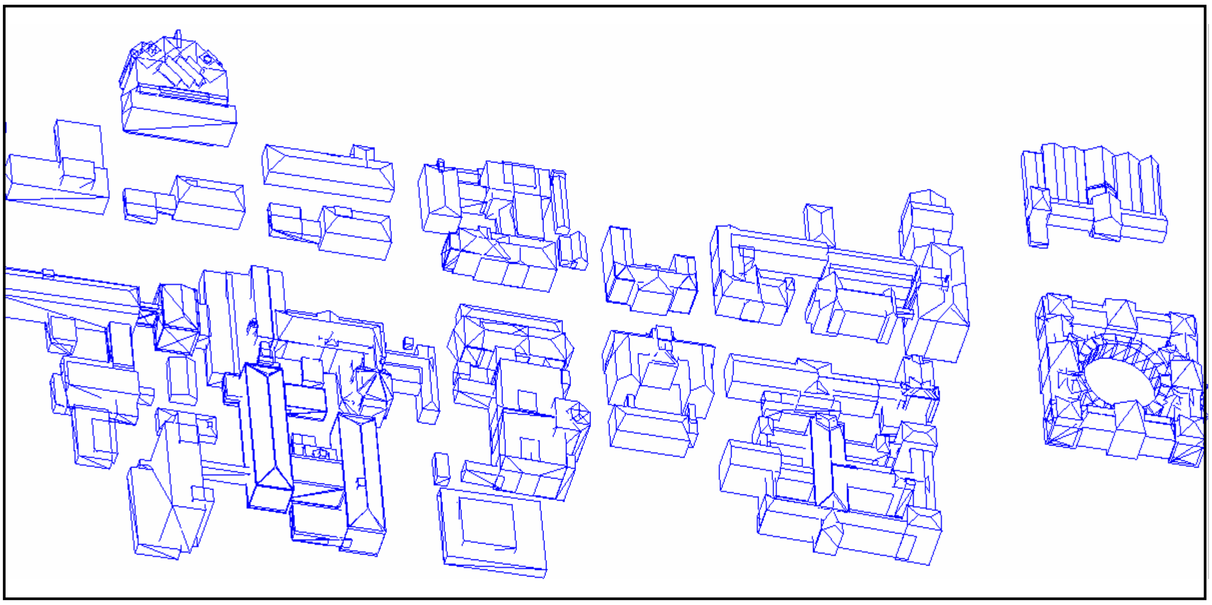


Figure 43: Nussalle area of city BONN (built up by the institut für photogrammetrie Bonn, universität Bonn)

As shown in the figure 43, most of the buildings of the Nussalle area of the city seem to be complex. In fact, it is found that they are formed due to the grouping of many simple buildings and not even a single complex building is found to have a single structure. It means these, in turn, are made of different simple buildings as shown in figure 44.

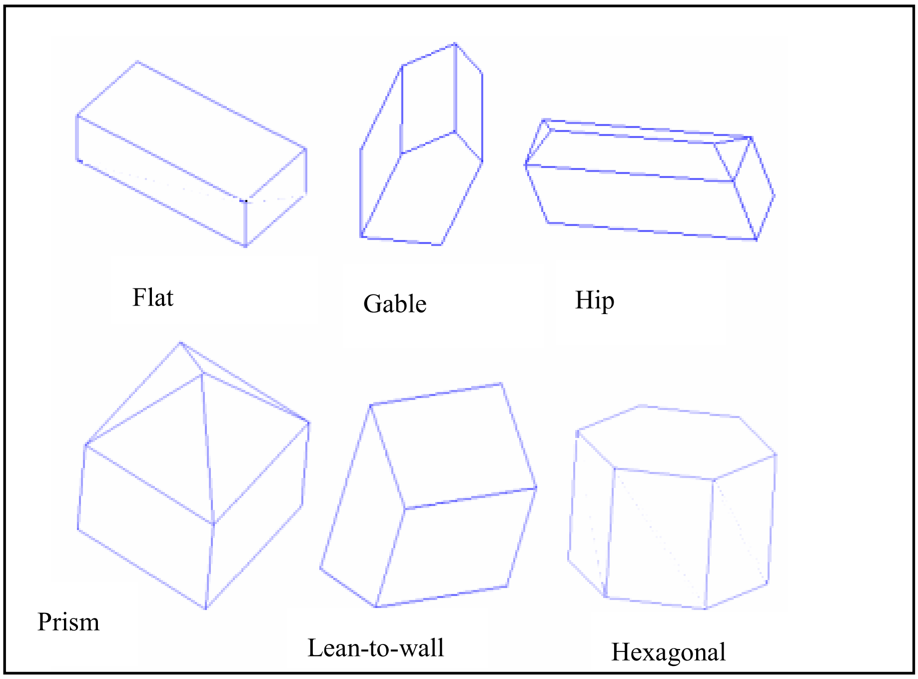


Figure 44: Different buildings

These simple buildings are named as Gable, Hip, Prism, Flat, Lean-to-wall and Hexagon. A complex building may be formed by a set of these simple buildings after their rotation, translation and scaling. It is therefore imperative to recognize individual simple buildings, which is then followed by the recognition of complex buildings.

5.3 Recognition of simple buildings

For the sake of understanding the whole process of recognition comprehensively, recognition of buildings is described before recognition of their roofs and ground plans. It is worth to mention that both of them have already been recognized using other existing building parameters. They will be described in brief thereafter. Neural Network tool "NeuNet", a free downloadable software package from (http://www.smartcode.com), is used for this purpose. Input data contains 250 different buildings and is stored in a *mdb* file as this software requires its inputs in this format. Since the existing data is in *sat* format, therefore a conversion utility has been developed to convert the *sat* format to *mdb* format. Out of them 100 buildings are selected for training the ANN and rest of the buildings are used for the prediction of results. A set of parameters is extracted for each building using its structure description studied in chapter 3 and is shown in table 7. These parameters are used as input for ANN. However as the ANN predicts the output from the given set of inputs but it is difficult to know how many and what kind of inputs it requires to give the desirable results. Therefore, an incremental approach of adding parameters to its input is selected to train it till it gives the acceptable output. In the beginning of the trial, only coordinates of the individual buildings are chosen as input but the output was not up to the desired mark. Therefore, other parameters are added step by step until a satisfactory result is obtained. Table 7 shows various parameters selected as possible inputs to the ANN:

Sr.no	Input	Description
i.	ID	Primary key field (0-250)
ii.	X,Y,Z	Coordinates of the building
iii.	GroundType	(1-square or rectangular, 2-hexagon)
iv.	TotalVertices	Total number of building vertices
v.	NoOfFaces	Total number of the faces of a building
vi.	FaceAngle	Angle between the roof faces, if any
vii.	RoofType	(1-flat,2-gable,3-hip,4-prism,5-hexagon,6-lean-to-wall)
viii.	BuildingId	(1-flat,2-gable,3-hip,4-prism,5-hexagon,6-lean-to-wall)

Table 7: Possible parameters as input to NN

The parameters shown in black have been used together for the successful predictions; parameters (in blue) are not used. As explained above, parameters are added incrementally after each training cycle until a satisfactory result is achieved. Further addition of remaining parameters is of no use and therefore is not used. Parameter in green is the output. Figure 45 shows the window for entering the parameters. Once the parameters are selected, the software reads data from the input file and display it as shown in figure 46. When the program begins to learn, it initializes its various parameters viz. *number of neuron* in hidden layers and maximum & minimum values of the other various inherent parameters. But for better results, these parameters are manually set by clicking the *advance* button in figure 45. A new window pops-up which displays these default values with the options to change them. These parameters are changed as per the requirement and depending upon the extreme values of parameters.

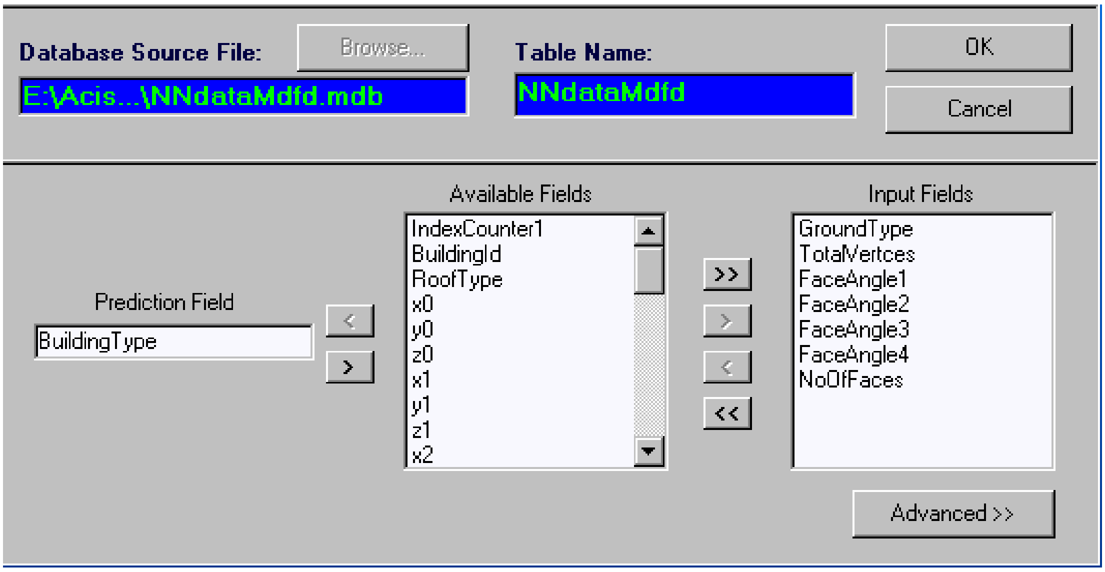


Figure 45: Input & output selection window

Another option, the software offers is that any number of parameters are added to the input but only selective parameters are chosen for the learning as shown under the *Available Fields* of the figure 45. It makes easier to add and remove various parameters while looking for the best possible combination.

BuildingType	GroundType	TotalVertces	FaceAngle1	FaceAngle2	FaceAngle3	FaceAngle4	NoOfFaces
2	1	10	19561	5911	5927	0	7
1	1	8	0	0	0	0	6
2	1	10	12590	6279	6279	0	7
1	1	8	0	0	0	0	6
2	1	10	9193	4588	4604	0	7
1	1	8	0	0	0	0	6
6	2	10	7889	7896	15817	0	7
5	1	12	0	0	0	0	8
1	1	8	0	0	0	0	6
5	1	12	0	0	0	0	8
1	1	8	0	0	0	0	6
5	1	12	0	0	0	0	8
1	1	8	0	0	0	0	6
2	1	10	8625	4313	4312	0	7
2	1	10	8609	4312	4297	0	7
2	1	10	8793	8387	14219	0	7
1	1	8	0	0	0	0	6
2	1	10	10637	9329	11433	0	7
2	1	10	10654	9312	11432	0	7
1	1	8	0	0	0	0	6
3	1	10	5005	5005	5004	5004	9
6	2	10	8083	7781	15897	0	7
1	1	8	0	0	0	0	6
1	1	8	0	0	0	0	6
3	1	10	5099	5099	3986	27413	9
1	1	8	0	0	0	0	6
4	1	10	11174	11174	5115	26284	9
1	1	8	0	0	0	0	6
3	1	10	11190	11191	5131	5131	9
1	1	8	0	0	0	0	6
5	1	12	0	0	0	0	8
1	1	8	0	0	0	0	6
2	1	10	6721	2546	4175	0	7
3	1	10	3850	3850	3879	27520	9
3	1	10	5453	5452	5652	25747	9
1	1	8	0	0	0	0	6
3	1	10	5100	5100	3986	27413	9
1	1	8	0	0	0	0	6
3	1	10	4824	4824	4920	26479	9
1	1	8	0	0	0	0	6
1	1	8	0	0	0	0	6
3	1	10	5471	5470	5272	5272	9
1	1	8	0	0	0	0	6
5	1	12	0	0	0	0	8
1	1	8	0	0	0	0	6
4	1	10	3307	3307	3288	28111	9
1	1	8	0	0	0	0	6

Figure 46: Screen shot of input parameters window

5.3.1 Splitting of input data

After initialization, ANN has to be trained properly to predict the desired results and for this input data has to be divided into training and test sets. For training set, about 100 data records are selected from the total of 250 records and remaining records are used to predict and test the result. The splitting of the data is shown in figure 47 below. Although more than 500 records of different buildings was available but only 250 records are chosen for the complete process. It is due to the fact that software only accepts a maximum of 250 records. However, these records are selected randomly from all the data available. It is possible to overlap the test and training data and is only desirable when the input data is not sufficient in records. It is required that training data should at least have minimum of 10 records of data for proper learning of the network but the test data can be as low as one record. On the other hand, too big data is also not desirable, as it will slow down the learning process. Therefore, intuitively an value of 100 records for training and 150 records for testing are taken here.

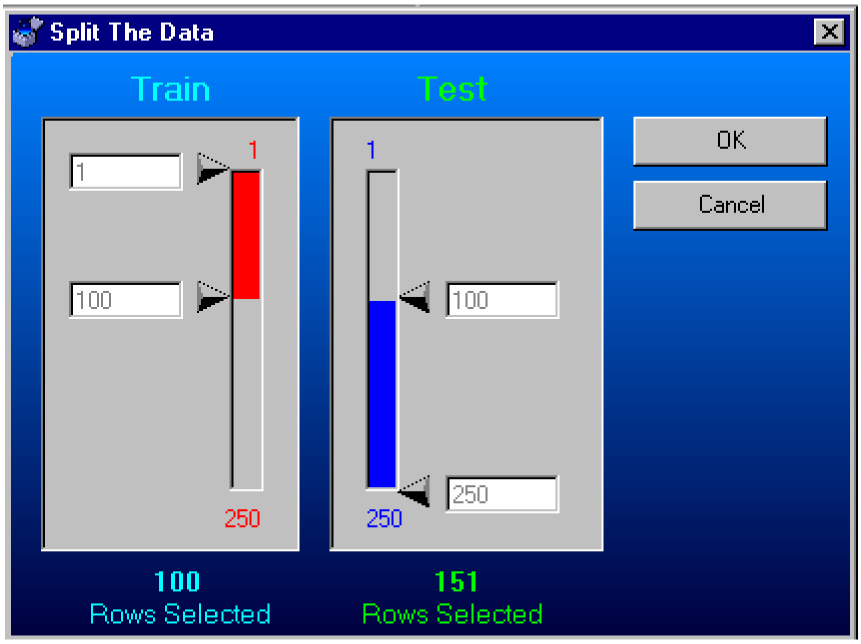


Figure 47: Training and testing pattern selection

5.3.2 Neural network training

It is necessary to initialize various control parameters of network before it begins learning (NeuNet 2001). Figure 48 shows these parameters and their explanation is as follows:

- *Learn rate* is rate is the rate at which network learns.

- *Momentum* is required to speed up calculations significantly which in turn helps in speeding the convergence and avoiding the local minima. It smoothens the weight changes and suppresses cross-stitching, that is cancels side-to-side oscillations across the error valley. When all weight changes are in the same direction the momentum amplifies the learning rate causing a faster convergence;

- *Verify rate*

This setting determines how many training cycles are made before a verify cycle is run. It is necessary to evaluate the current network and report the error. The best setting depends on the speed of computer and the number of records and fields in your training set.

- *Best Error*

This error indicates the Root Mean Square (RMS) error for the best verify cycle thus far. It is also called "Standard Error of Estimate" and is calculated as SquareRoot {SumOfAll[(Actual-Predicted)^2] / NumberOfPredictions}. As this calculation is performed using normalized values, so it may be stated as percent. The blue coloring on the **History Graph** shows which previous cycle produced the **Best Error**.

- *Current Error*

This error indicates the Root Mean Square (RMS) error for the most recent verify and is called "Standard Error of Estimate". It is calculated as SquareRoot of {SumOfAll[(Actual-Predicted)^2]/NumberOfPredictions}. As this calculation is performed using normalized values, so it may be stated as percent. The number that appears in this box is constantly graphed in the **History Graph**.

- *History of Error Graph*

This graph shows a history of the prediction error achieved during the previous verify cycles. The blue coloring marks which previous cycle was saved as the best.

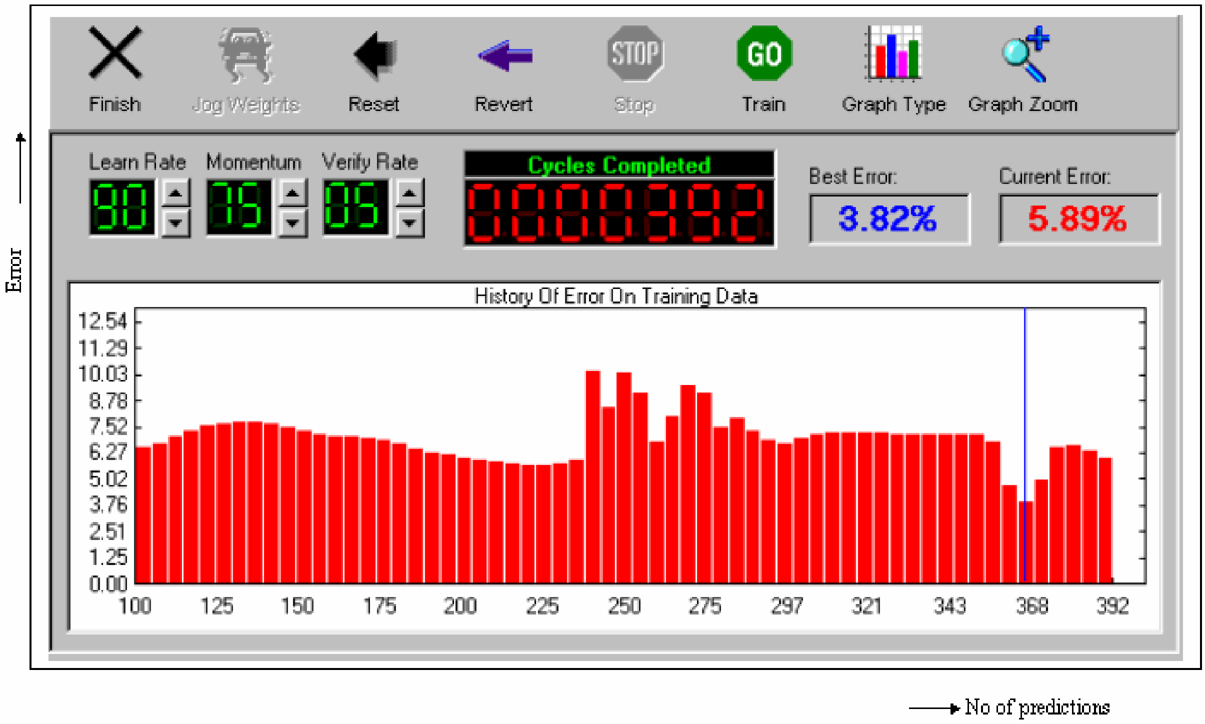


Figure 48: History graph of prediction errors

Training begins with all weights set to random numbers. For each data record, the predicted value is compared to the desired (actual) value and the weights are adjusted to move the prediction closer to the desired value. Many cycles are made through the entire set of training data with the weights being continually adjusted to produce predictions that are more accurate. Learning of the network is controlled by setting three parameters viz. *learn rate, momentum and verify rate* as shown in *history graph* in figure 48. This graph shows a history of the prediction error achieved during the previous verify cycles. *Learn rate* and *momentum* are best set during training run in the beginning and it is better to set *learn rate* always greater than *momentum*. If *learn rate* and *momentum* are set too low, the training will be very slow with a smooth, gradual improvement. On the other hand if *learn rate* and *momentum* are set too high, the training will be very choppy, and chaotic. *Verify rate* determines how many training cycles are made before a verify cycle is run and is necessary to evaluate the current network and report the error. Once the training begins, it starts giving current status of the training by reporting *current error* and *best error*. Best error indicates the Root Mean Square (RMS) error for the best verify cycle thus far and is also called "Standard Error of Estimate" whereas *current error* indicates the Root Mean Square (RMS) error for the most recent verify.

5.3.3 Scatter graph

This graph provides both numeric and graphical report showing the accuracy of network predictions (figure 49).

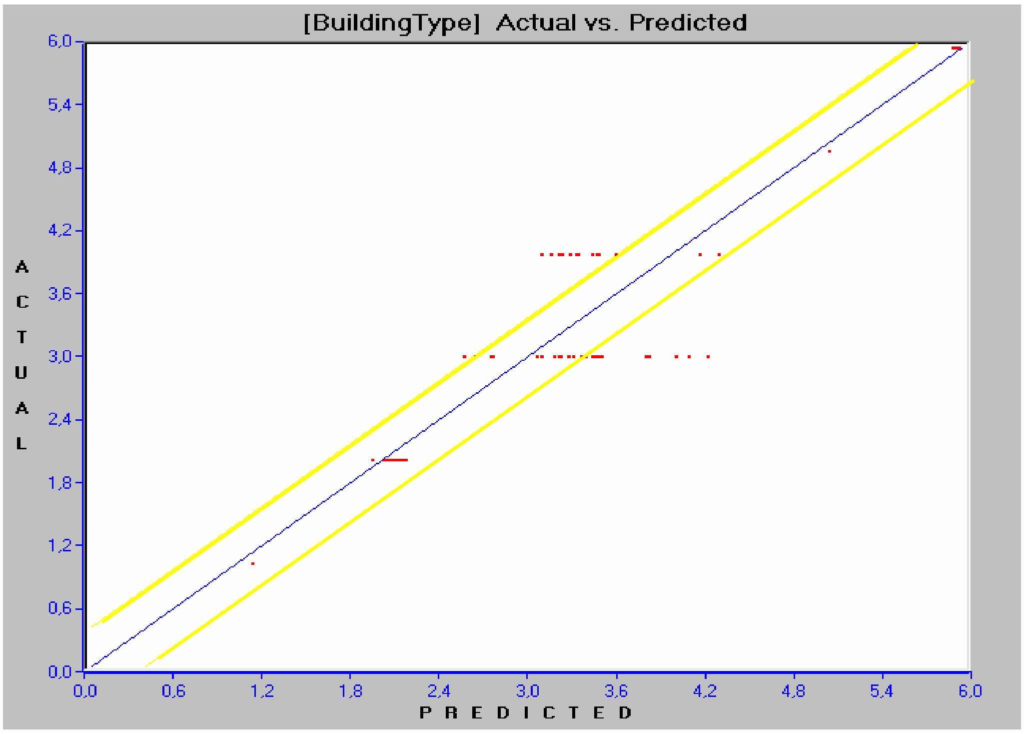


Figure 49: Scatter graph

The closer the scattering (red dots) is to the blue diagonal line, the more accurate are the predictions. The band shown by faint yellow lines indicates a certainty band, as defined by the RMS Error. Approximately 2/3 of the testing set should lie within this certainty band. Correlation coefficient is a number between zero and one that indicates how well the prediction is correlated to the actual. A value of one indicates perfect prediction whereas value of zero indicates no relationship between prediction and target.

5.3.4 Predictions

Once the learning is over, prediction results can be seen immediately as shown in figure 50. It has completed 23000 cycles to predict these results and has taken a total time of approximately 3 seconds on MS Window platform with PENTIUM III machine.

It shows that prediction results are very good and close to the expectations for building types 1,2,5 and 6. However there is a little confusion in predicting two building types (i.e. hip and prism). These two buildings have more in common (i.e. ground plan, number of vertices, and number of faces) and are different in only their face angles among the input parameters. This problem can be solved by either of the following two ways. Firstly it may be achieved by changing the various control parameters i.e. momentum, learning rate and verify rate and allowing more iteration to learn it better. However, it has been shown that by doing so, network goes to over-learning stage, which is not desirable. Second way of improving the performance is by introducing more input parameters so that variations become more among them. Experiment has shown that when the additional parameter, roof type, is added and the whole process was repeated by keeping the control parameter unchanged, results have been excellent. Following figure however shows result without adding roof types and kept purposefully to convey the importance of adding additional parameters.

BuildingType	Predicted	Difference	GroundType	TotalVertces	FaceAngle1	FaceAngle2	FaceAngle3	FaceAngle4	NoOfFaces
3	3,38	0,38	1	10	9795	9789	9067	22332	9
2	2,17	0,17	1	10	7676	3830	3846	0	7
2	2,17	0,17	1	10	7691	3846	3845	0	7
2	2,15	0,15	1	10	10335	5151	5151	0	7
1	1,12	0,12	1	8	0	0	0	0	6
3	3,45	0,45	1	10	6135	6139	6038	25361	9
1	1,12	0,12	1	8	0	0	0	0	6
3	3,11	0,11	1	10	3846	3846	3717	3717	9
2	1,94	0,06	1	10	14935	6177	10286	0	7
4	3,17	0,83	1	10	3329	3329	2514	2514	9
3	3,41	0,41	1	10	8046	8046	7588	23811	9
2	2,03	0,03	1	10	18290	6562	6546	0	7
2	2,04	0,04	1	10	17949	6732	6717	0	7
3	3,48	0,48	1	10	4842	4842	5598	25801	9
3	3,48	0,48	1	10	7330	7330	6074	25325	9
2	2,13	0,13	1	10	12063	6031	6032	0	7
4	3,45	0,55	1	10	6869	6869	6361	6362	9
2	2,08	0,08	1	10	14319	6659	7628	0	7
4	3,45	0,55	1	10	6135	6139	6038	25361	9
6	5,96	0,04	2	10	9931	5744	15707	0	7
2	2,17	0,17	1	10	7691	3846	3845	0	7
6	5,96	0,04	2	10	9930	5745	15707	0	7
2	2,10	0,10	1	10	14777	7368	7377	0	7
2	2,14	0,14	1	10	11150	5575	5575	0	7
2	2,17	0,17	1	10	7676	3830	3846	0	7
6	5,97	0,03	2	10	8902	6789	15707	0	7
4	3,34	0,66	1	10	3329	3328	3840	3839	9
2	2,13	0,13	1	10	12179	6089	6089	0	7
5	5,08	0,08	1	12	0	0	0	0	8
1	1,12	0,12	1	8	0	0	0	0	6
2	2,15	0,15	1	10	10335	5151	5151	0	7
2	2,15	0,15	1	10	10335	5151	5151	0	7
1	1,12	0,12	1	8	0	0	0	0	6
3	3,45	0,45	1	10	6135	6139	6038	25361	9
1	1,12	0,12	1	8	0	0	0	0	6
5	5,08	0,08	1	12	0	0	0	0	8
2	2,14	0,14	1	10	11153	5761	5392	0	7
1	1,12	0,12	1	8	0	0	0	0	6
3	3,51	0,51	1	10	3746	3747	4233	27166	9
3	3,51	0,51	1	10	3746	3747	4233	27166	9
1	1,12	0,12	1	8	0	0	0	0	6
2	2,10	0,10	1	10	14551	7260	7259	0	7
2	2,06	0,06	1	10	16868	7257	7273	0	7
1	1,12	0,12	1	8	0	0	0	0	6
2	2,15	0,15	1	10	10161	5072	5088	0	7
3	3,40	0,40	1	10	2747	2749	2461	28938	9

Figure 50: Prediction results of building types

It completes the whole cycle of the learning and predicting of different building types. The complete process is described in detail for the recognition of buildings. As the recognition of roofs and ground plan will follow the same principle, therefore, it will be described in brief and the more emphasis will be given to the results.

5.4 Recognition of roofs

In recognition of roof, same data is taken but different set of parameters is chosen. Following parameters in table 8 are the possible inputs and due to the reason stated above, only parameters in black are selected as inputs:

Sr. No	Input	Description
1.	X,Y,Z	Coordinates of the building
2.	GroundType	(1-square or rectangular, 2-hexagon)

3.	TotalVertices	Total number of building vertices
4.	NoOfRoofFaces	Total number of the faces of a roof
5.	FaceAngle	Angle between the roof faces, if any
6.	RoofType	(1-flat,2-gable,3-hip,4-prism,5-hexagon,6-lean-to-wall)
7.	BuildingId	(1-flat,2-gable,3-hip,4-prism,5-hexagon,6-lean-to-wall)

Table 8: Building parameters for roof type recognition

After training the ANN with above parameters, it has been tested with different set of roof styles. Figure 51 shows the results for recognition of various roof styles.

RoofType	Predicted	Difference	FaceAngle1	FaceAngle2	FaceAngle3	FaceAngle4	x0	y0	z0	x1	y1
1	1,01	0,01	0	0	0	0	186257	-97220	-14791	185940	-99037
1	1,01	0,01	0	0	0	0	172029	-69230	-12514	171266	-66652
6	5,99	0,01	0	0	0	0	178121	-169827	-16602	178179	-170022
1	1,01	0,01	0	0	0	0	185387	-101066	-13883	183739	-110520
1	1,01	0,01	0	0	0	0	153179	54229	-13727	151701	46240
1	1,01	0,01	0	0	0	0	162658	-48574	-14835	163391	-44409
3	3,01	0,01	7593	7588	6056	6050	174511	-246146	-12430	175639	-249989
1	1,02	0,02	0	0	0	0	166514	-29807	-15321	165372	-30622
3	3,02	0,02	7901	7900	7847	7846	160746	-108678	-13458	161659	-111610
2	2,02	0,02	11150	5575	5575	0	182677	-249793	-12092	179407	-268080
2	1,98	0,02	14319	6659	7628	0	169912	-159507	-12461	174871	-176329
3	3,04	0,04	6002	6002	5846	5845	159079	-71007	-12790	159984	-65834
2	1,96	0,04	9671	4894	4776	0	175113	-117735	-13128	178798	-130295
6	5,96	0,04	0	0	0	0	154979	-49692	-13305	156442	-47099
6	5,95	0,05	0	0	0	0	159895	-65465	-13305	161298	-62943
3	2,95	0,05	9789	9789	8987	22412	177890	-75462	-12601	177259	-73424
3	3,05	0,05	9380	9369	9707	9717	191687	-166909	-13815	191045	-170390
6	5,94	0,06	0	0	0	0	176300	-77995	-12682	175920	-78232
3	2,94	0,06	5560	5561	3581	3581	190136	-171432	-14219	188736	-174670
2	1,94	0,06	10486	5227	5227	0	196969	-189212	-14022	192320	-215810
4	4,07	0,07	7086	7086	9351	9350	154295	3042	-13591	152847	7935
3	2,90	0,10	5565	5561	3577	3581	193805	-205170	-14219	192404	-208409
3	3,12	0,12	7724	7724	5875	5875	157301	-9981	-12784	156273	-6517
3	2,85	0,15	5553	5568	4033	4033	195326	-192747	-14219	195100	-198667
4	3,85	0,15	13206	13206	6248	25151	154261	3252	-13591	150868	587
2	2,16	0,16	10358	5179	5179	0	150532	-121389	-12031	151251	-123882
2	1,83	0,17	7691	3846	3845	0	178391	-235690	-12170	175123	-253977
4	4,17	0,17	10154	10154	10044	21355	193032	-177209	-14646	194838	-183289
3	2,81	0,19	3746	3747	4233	27166	167853	-81555	-14040	166116	-75749
4	3,80	0,20	8844	8839	8717	8721	188949	-215861	-14628	190421	-207792
2	2,20	0,20	10335	5151	5151	0	179742	-250364	-12365	177663	-261894
4	4,21	0,21	9342	9342	9220	22179	193754	-205940	-14628	195659	-212107
2	1,78	0,22	7676	3830	3846	0	179653	-239864	-12170	176383	-258150
3	2,74	0,26	9396	9396	7956	7956	192167	-163868	-13815	191246	-168855
2	2,26	0,26	14415	8483	8500	0	176419	-148997	-14436	175005	-156431
2	2,26	0,26	8251	4116	4135	0	185088	-202180	-14022	191165	-222707
2	2,26	0,26	8251	4116	4135	0	185088	-202180	-14022	191165	-222707
3	2,73	0,27	3746	3747	4233	27166	167853	-81555	-13770	166116	-75749
4	3,73	0,27	6853	6854	6345	25054	161119	-99231	-13784	159917	-95087
2	2,28	0,28	16868	7257	7273	0	168999	-93482	-13928	171236	-81079
2	1,72	0,28	14546	6732	7781	0	155759	-107474	-12017	157217	-99617
2	2,30	0,30	9460	4721	4738	0	151646	-121538	-12031	154510	-106292
2	2,31	0,31	10468	5224	5243	0	169749	-101695	-13128	174891	-118887
2	1,69	0,31	12063	6031	6032	0	182907	-106704	-12361	175976	-84631
2	1,69	0,31	19036	6170	6193	0	183322	-87424	-14101	181747	-82369
5	5,32	0,32	8902	6789	15707	0	181520	-256690	-12340	182456	-251767

Figure 51: Prediction results

5.5 Recognition of ground plan

The current data contains buildings with only two ground types viz. rectangular and hexagonal. To recognize them, only the coordinates of the buildings are used as shown in figure 52.

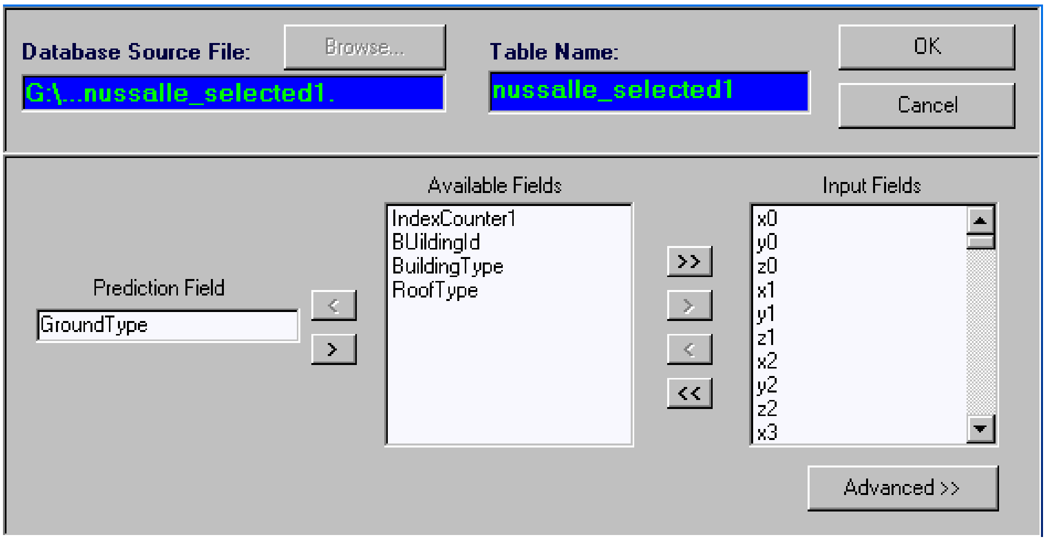


Figure 52: Input & prediction parameters for the recognition of ground plan

After training the ANN with above parameters, it has been tested with different set of ground plans. Results are obtained as shown in figure 53.

GroundType	Predicted	Difference	x0	y0	z0	x1	y1	z1	x2
1	1,001	0,001	1062	-62	-361	1070	-62	-367	1070
1	1,001	0,001	1099	-62	-352	1094	-62	-348	1094
1	1,001	0,001	1099	-62	-352	1094	-62	-348	1094
1	1,015	0,015	759	-58	-59	759	-75	-59	759
2	1,966	0,034	765	-72	-92	767	-72	-91	767
2	1,966	0,034	745	-75	-76	746	-75	-74	746
1	1,016	0,016	741	-49	-86	741	-61	-86	741
2	1,966	0,034	765	-72	-92	767	-72	-91	767
1	1,014	0,014	756	-49	-93	756	-73	-93	756
1	1,014	0,014	751	-49	-84	751	-73	-84	751
1	1,013	0,013	772	-49	-79	771	-49	-75	771
1	1,014	0,014	763	-49	-71	762	-49	-67	762
1	1,011	0,011	768	-49	-82	767	-49	-78	767
1	1,001	0,001	1099	-62	-352	1088	-62	-343	1088
1	1,001	0,001	1090	-62	-383	1097	-62	-388	1097
1	1,005	0,005	1105	-60	-106	1079	-60	-85	1079
1	1,004	0,004	1069	-60	-141	1080	-60	-126	1080
1	1,005	0,005	1051	-60	-122	1057	-60	-115	1057
1	1,005	0,005	1068	-60	-102	1082	-60	-82	1082
1	1,002	0,002	1051	-59	-252	1057	-59	-243	1057
1	1,001	0,001	1080	-62	-320	1073	-62	-330	1073
2	1,947	0,053	1073	-89	-237	1073	-89	-238	1073
1	1,001	0,001	1080	-62	-320	1073	-62	-330	1073
1	1,001	0,001	1099	-62	-352	1094	-62	-348	1094
1	1,001	0,001	1099	-62	-352	1094	-62	-348	1094
1	1,001	0,001	1055	-62	-355	1062	-62	-361	1062
1	1,001	0,001	1080	-62	-375	1090	-62	-383	1090
1	1,013	0,013	764	-49	-71	763	-49	-67	763
1	1,001	0,001	1070	-62	-366	1081	-62	-375	1081
1	1,001	0,001	1055	-62	-355	1062	-62	-361	1062
1	1,001	0,001	1099	-62	-352	1094	-62	-348	1094
1	1,008	0,008	909	-60	1	917	-60	10	917
1	1,005	0,005	975	-60	-21	970	-60	-27	970
1	1,027	0,027	615	-61	0	623	-61	6	623
1	1,025	0,025	726	-58	98	726	-86	98	726
2	1,966	0,034	933	-71	-69	942	-71	-65	942
1	1,043	0,043	776	-58	124	768	-58	130	768
1	1,049	0,049	758	-58	110	763	-58	116	763
1	1,001	0,001	1062	-62	-361	1070	-62	-367	1070
1	1,007	0,007	904	-61	-107	909	-61	-100	909
1	1,004	0,004	897	-60	-128	897	-68	-128	897
1	1,005	0,005	883	-60	-144	887	-60	-139	887
1	1,004	0,004	900	-61	-133	900	-68	-133	900
1	1,007	0,007	914	-61	-94	942	-61	-115	942
1	1,007	0,007	906	-61	-89	924	-61	-104	924
1	1,005	0,005	901	-60	-89	901	-73	-89	901

Figure 53: Prediction results of ground plan

5.6 Recognition of complex buildings

Complex buildings as described in chapter 3, are composed of simple buildings, as these buildings are located in so close vicinity that whole group forms a complex structure. Figure 54 shows a complex building that is formed from more than 30 simple buildings shown in figure 43.

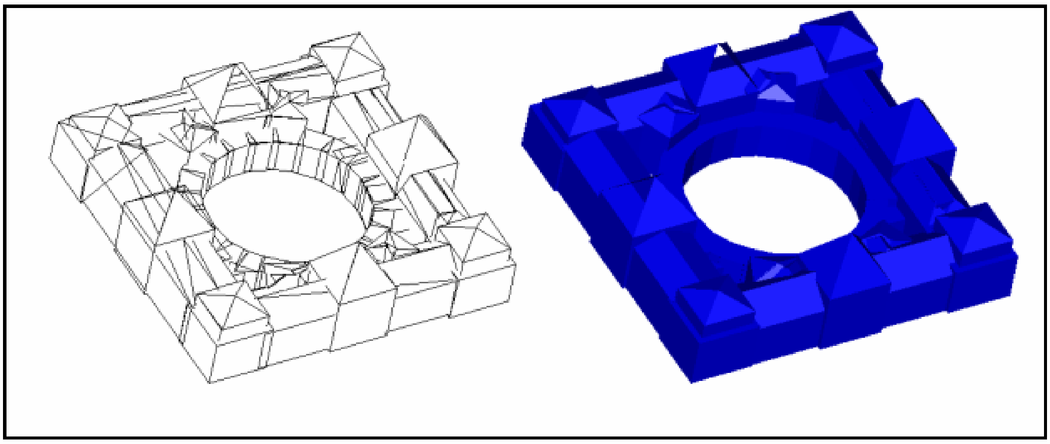


Figure 54: A complex building (wire frame & solid)

Recognition of such buildings is a tedious task as it involves the extraction of complete information of simple buildings joined to form these complex buildings. Fortunately complete information about the individual buildings is already available as described above in *recognition of simple buildings* but the difficult task still remains as how to get the identification of participating buildings. Well known algorithm like Delanauey Triangulation have been applied unsuccessfully by taking centers of the simple buildings as its input. Since these buildings are of various length and shape, difficulty was faced in deciding the cut-off value of edge distance. Too small value gave more than one cluster of a complex building and too large value resulted in cluster of simple buildings comprising complex building plus few others near by buildings.

A new algorithm is developed which takes care of the shortcoming of the above algorithm. It starts with a single building and finds its immediate neighbors. Once the immediate neighbors are found, next step involves the finding of immediate neighbor of these newly found neighbors and the process repeats until there is no more neighbor. It gives rise to the cluster of simple buildings forming a complex building. The whole process repeats itself for the next cluster until there is no more clusters left. A flowchart of the algorithm is shown in figure 56. This algorithm has been implemented in C++ under the Window environment. To test it, again the same set of buildings of city BONN is used as input. Figure 55 shows some of the important clusters obtained.

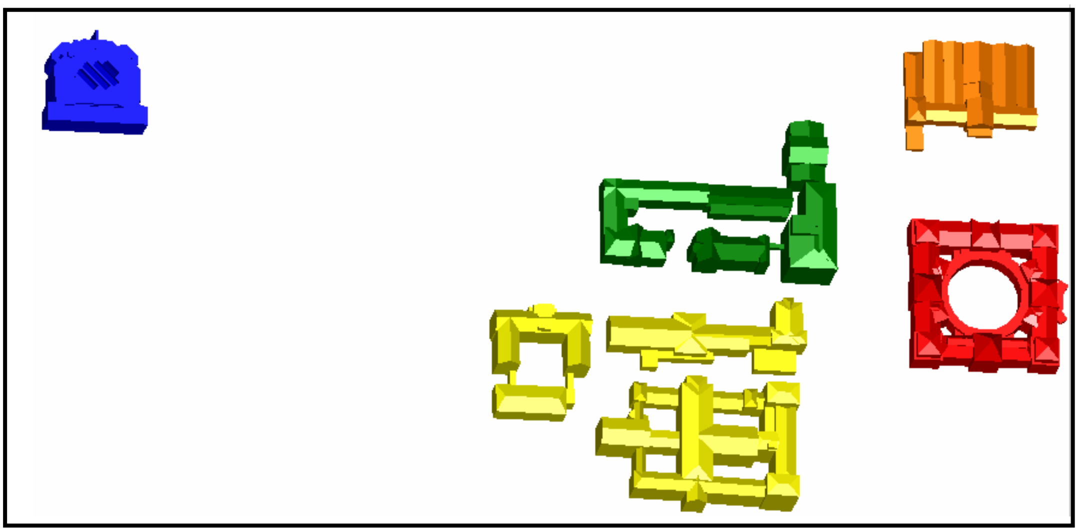


Figure 55: A clustered buildings

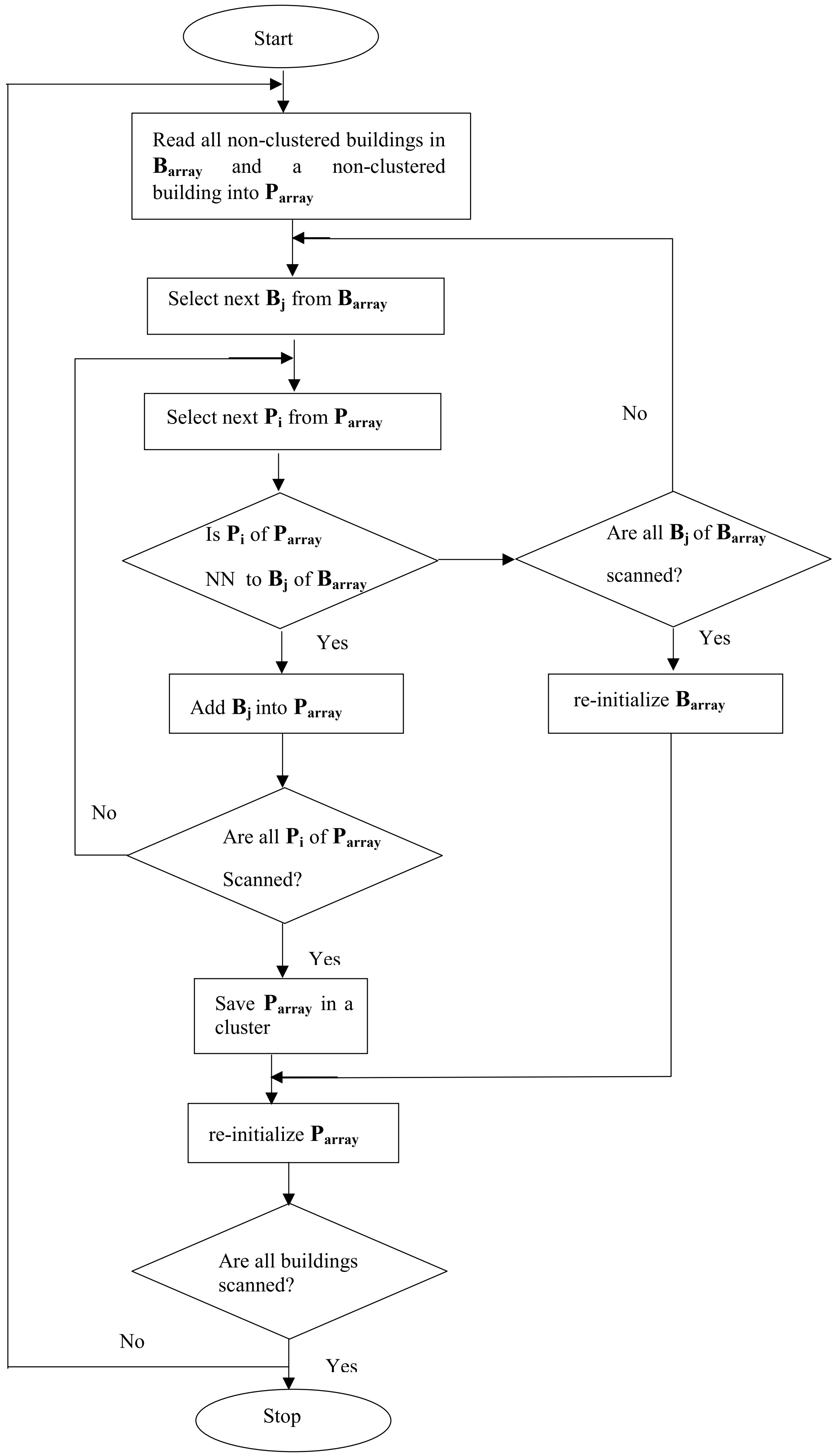


Figure 56: Clustered algorithm

Each color represents a cluster indicating a complex building. As the complete information of the participating buildings is known now, the next step is to remove the redundant information in them. For example, two building which are very close, the two intersecting faces (walls) have the same vertices and therefore redundant vertices has to be removed. Secondly, as only the outer faces are required, any inner face if any, has to be removed. It is done by applying convex hull to the entire complex buildings so that just the outer vertices are found. Therefore, convex hull algorithm is applied and figure 57 shows a convex hull wire frame for one of the above building.

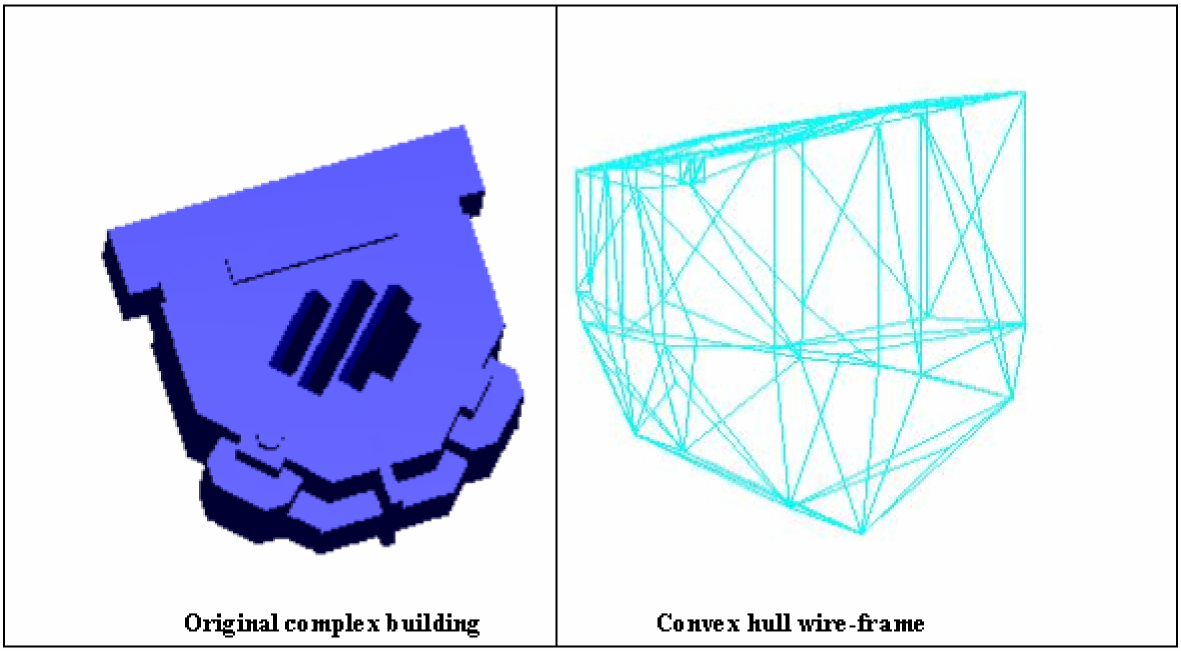


Figure 57: Convex hull

Other parameters, which are known now and can be applied as inputs to ANN are:

- Ground plan
- Various roof heights
- Roof styles
- Roof face angles
- Building area
- Number of wall faces
- Number of roof faces
- Number of ground faces

These parameters may be added systematically and gradually until the best possible result is found in a manner described above. However, the lack of sufficient amount of data, it is not possible to carry out the process for complex building recognitions.

A hierarchical approach for 3D building recognition using ANN is successfully studied here. The next chapter is focused on study of rules and constraints of 3D aggregation. A different set of constraints and consequently is formed based on the findings done in the preceding chapters.

Chapter 6

3D structure recognition at cluster level

As stated earlier in the preceding chapters, a 3D city model mostly consists of buildings and roads. These buildings are orderly arranged along the roads and therefore may be perceived as a whole immediately, but are not explicitly represented as an internal description of graphics. These non-local structures may be salient and of strong visual appeal, thus they seem to be of some cognitive relevance to a viewer. It is, therefore desirable to develop methods to detect these structures automatically in order to support their generalization. Following sections study two approaches for determining the 3D structure of objects that appears to have formed a group.

6.1 Structure recognition using cluster algorithm

A clustering algorithm is usually applied to determine the macro level structure. Among the various algorithm techniques, Minimum Spanning Tree (MST) (Ahuja 1993) is a suitable. It takes a graph consisting of weighted edges and nodes as input and result in minimum weighted sum of edges. This set of edges contains long as well as small edges. The deletion of large edges results in disjoint sets of nearest buildings.

The center of a ground plan of the building is treated as node and the link between various nodes serves as edges. MST algorithm is applied twice to obtain the desire clustering. First MST algorithm is used to obtain cluster based on spatial proximity.

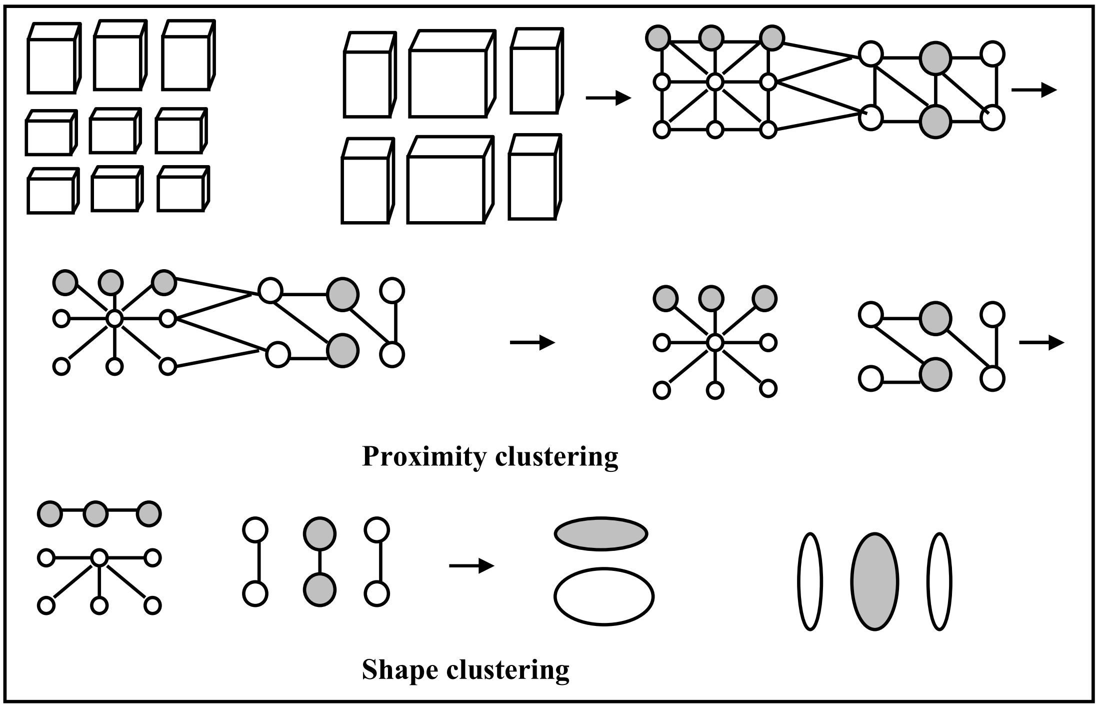


Figure 58: Neighborhood relations

Prior requirement of the MST is the information about the nearest neighborhood, which can be obtained through Delauney Triangulation method.

MST is then executed second time to obtain the set of minimum weighted edges based upon similarity. Long edges are then removed to obtain the spatial clusters. Size (or shape) based clustering is obtained by running the MST second time. Similarly, any other parameter described in chapter 3, such as color, texture, height of the buildings can be a criterion for second time MST to find the required clusters. A complete sequence of operations is shown in figure 58.

6.2 Structure recognition using perceptual grouping

Perceptual grouping is very important for object recognition and human visual system uses wide variety of perceptual grouping mechanism to describe its structure. Even with no high level or semantic knowledge available, the human visual system spontaneously organizes elements of the visual field. It happens due to the grouping of low-level features leading to a higher-level structure. These higher-level structures may be further combined to yield another level of higher-level structures and this process may be repeated until a meaningful semantic representation is achieved that may be used by a higher-level reasoning process. In the absence of information for perceptual grouping, it is difficult for humans to make an intelligent decision regarding the structure or recognition of an object. These phenomena have been studied by Gestalt psychologists who observed how some arrangements of picture elements tend to be seen as 'belonging together', thereby forming natural groups. Often these may appear to stand out from the surrounding elements, i.e. as 'figures' against 'grounds' (Thomson 2000).

Direct analogies have therefore been identified between the Gestalt perceptual grouping principles and the procedures required for successful map generalization. This is especially so, since they provide a means of predicting what the result of a generalization process should look like so that it can be readily comprehended by a map user (DeLucia 1987). 3D structure recognition is not dependent upon only one process of perceptual grouping but must use the hierarchy of perceptual grouping processes based upon following concepts:

i. *Grouping by proximity*: Similar objects that are close together in space appear to belong together and tend to be perceived together. In a given city model, most of the buildings are situated very close. These buildings, when viewed from a certain height above the ground, tend to form a particular shape. These buildings should be clustered together.

Most of these buildings are located along the roads. In fact, there is hardly any building that is not connected by a road. This property of the city model is exploited here for the perceptual grouping. This exercise gives different clusters as the whole buildings in the city are divided into groups pertaining to road proximity. Once these clusters, named *first level* clusters, are obtained, various grouping principles are applied to them.

Once a *first level* cluster of buildings along a given road is detected, then second *level* clusters based upon proximity among buildings can be found.

Proximity measurement: Proximity of the buildings can be easily found by using Delaneuay triangulation method. Another method, an easily computable, is used as follow:

If d_{min} is the distance between two buildings and s_{min1} and s_{min2} are the lengths of shortest sides of the buildings (figure 59), then measure of proximity can be defined as

$$M_{prox} = 1 - min\ ((1,\ d_{min}/min(s_{min1},\ s_{min2}))$$

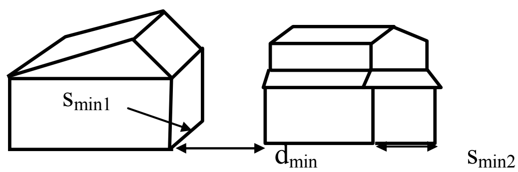


Figure 59: Proximity measurement

and always has value between zero and one. Group of buildings along a given road and satisfying a given value of $\mathbf{M}_{prox}$ form a cluster.

Figure 60 shows a cluster of buildings with different shapes along the roads. It gives rise to four clusters of buildings lying either side of the roads. Once these cluster are formed, principle of proximity is applied which give rise to second level cluster. These clusters, once recognized, can be used for generalization. For example, buildings lying within a specified distance are clustered as shown by encircled boundaries in figure 60 may be generalized.

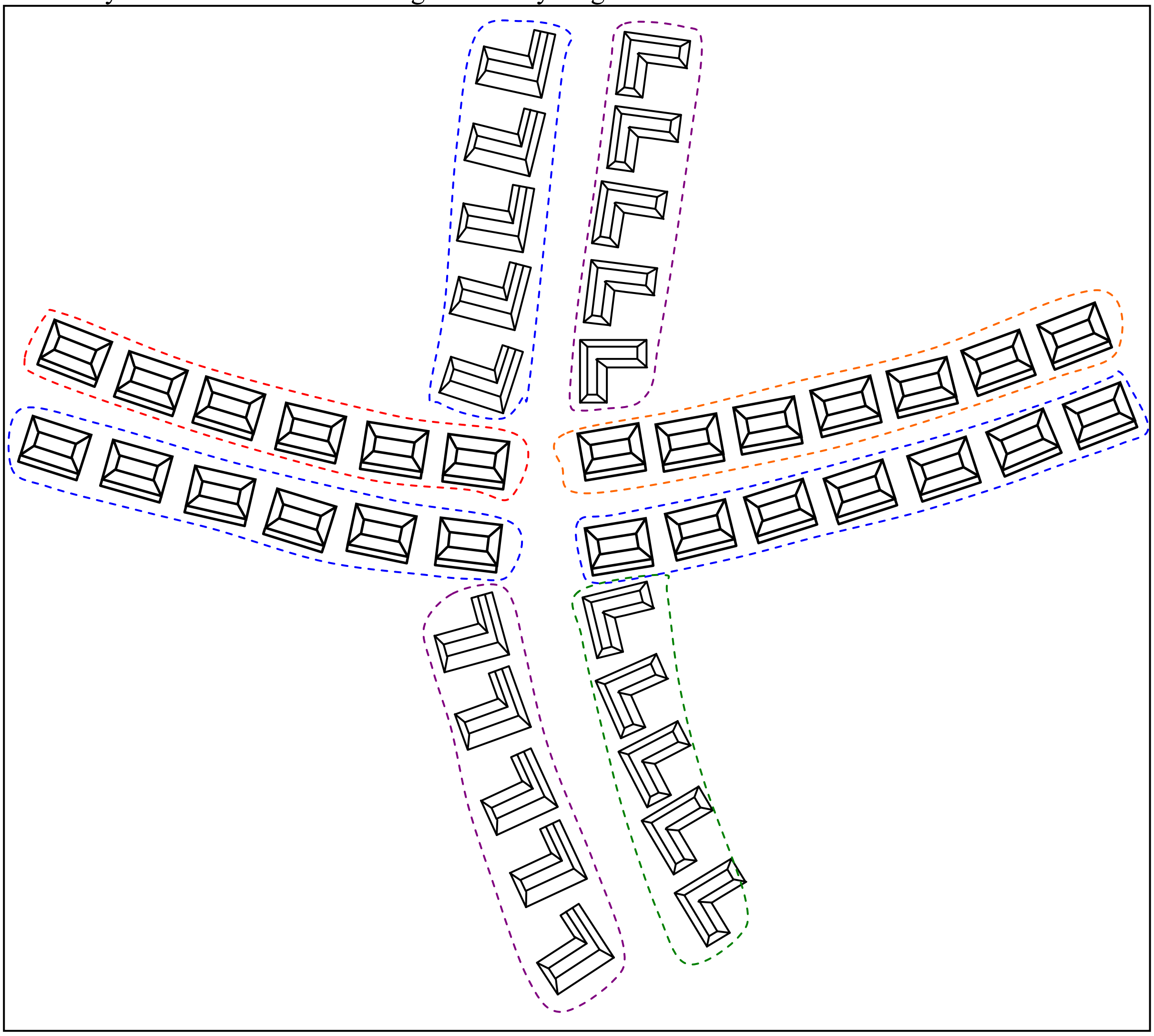


Figure 60: **Grouping by proximity**

ii. *Grouping by continuity*: There is a tendency in our perception to follow a direction, to connect the elements in a way that makes them seem contiguous or flowing in a particular direction as a line or a smooth curve (Brassard 2000). Therefore the principle of perceptual of good continuation demands that map symbols that appear to follow in the same direction (as in a straight line or simple curve) should be grouped together.“ Here again cluster of buildings along roads are taken as shown in figure 61. Different clusters are formed. After applying the principle of continuation and without violating other constraints, each cluster may be generalized to a single curved building block.

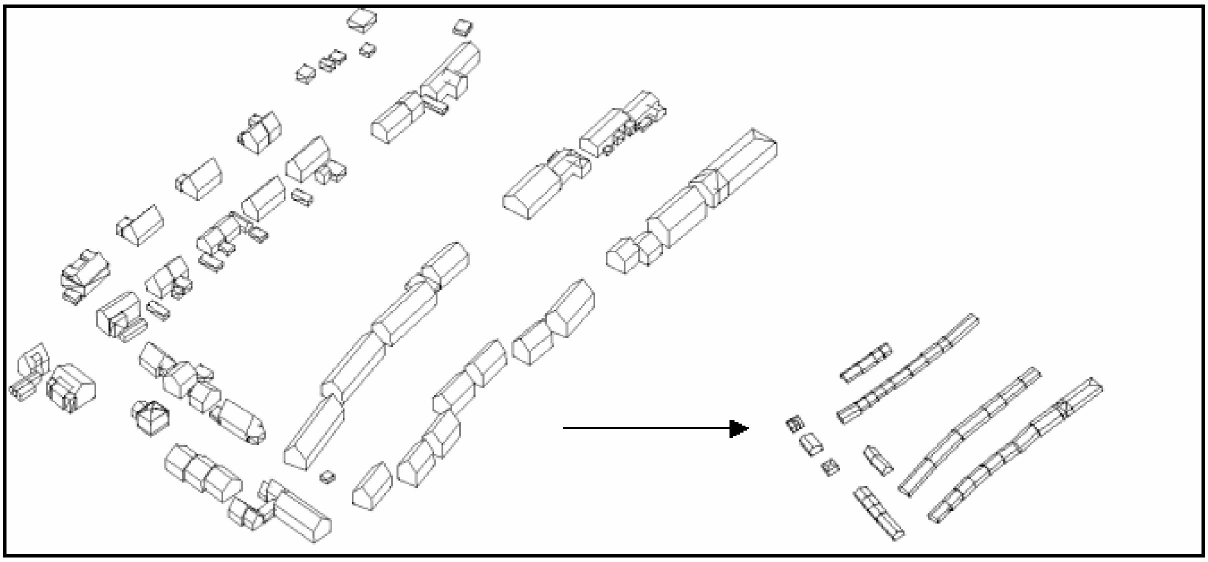


Figure 61: The principle of continuation

Continuity measurement: is used to identify all the buildings along a given side of the road and draw a line segment with length equal to the length of the building and orientation similar to the orientation of building as shown in figure 62. This gives us a number of segments as shown below:

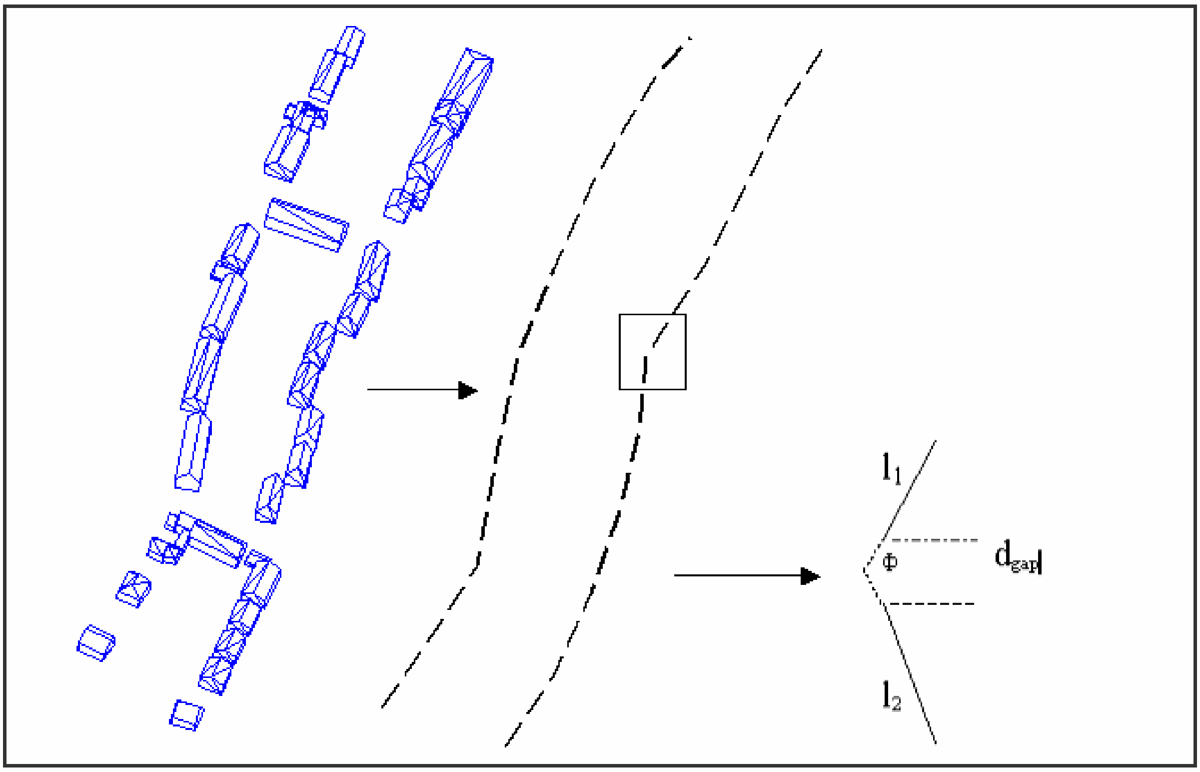


Figure 62: The principle of continuity measurement

If l_1 and l_2 are the length of two segments and d_{gap} is the gap between them and Φ is the angle between them, then the significance of continuation, M_{con}, is given by

$$M_{con} = min\left(\frac{(l_1 + l_2)}{(l_1 + l_2 + d_{gap})}, \cos^2\Phi\right)$$

$COS^2(\Phi)$ is equal to one when both the lines are perfectly continuous and gradually reduces to zero as the line segments become discontinuous.

iii. *Grouping by parallelism*: Buildings lying along both sides of a given road are clustered together. A pair of building on either side of the road is taken and their parallelism is calculated as follow:

Let δ_{max} is the maximum width and δmin is the minimum width between the two closest faces of two nearby buildings as shown in figure 63, then measure of parallelism

$$M_{pas} = 1 - \min((1, \delta_{min}/\delta_{max}))$$

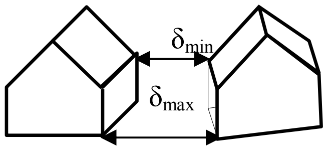


Figure 63: Principle of parallelism

A sum of all values M_{pas} divided by total number of pairs gives the quality of measurement. Comparisons of quality of parallelism after and before generalization reflect the quality of generalization. Upon generalization and applying the principle of parallelism, we get the result as shown in figure 64. Generalized buildings are still parallel to the roads

iv. *Grouping by similarity*: Similar objects tend to be seen together as forming a group (Brassard 2000). Figure 64 shows four rows of different buildings. These buildings may have same roof style, color or texture. Different clustered may be obtained based upon either of these attributes. Figure 64 shows the cluster obtained via similarity. The resulting clusters bounded by dotted colored boundaries are also shown there in, where the aggregation is among the buildings along rows and column as well may be done.

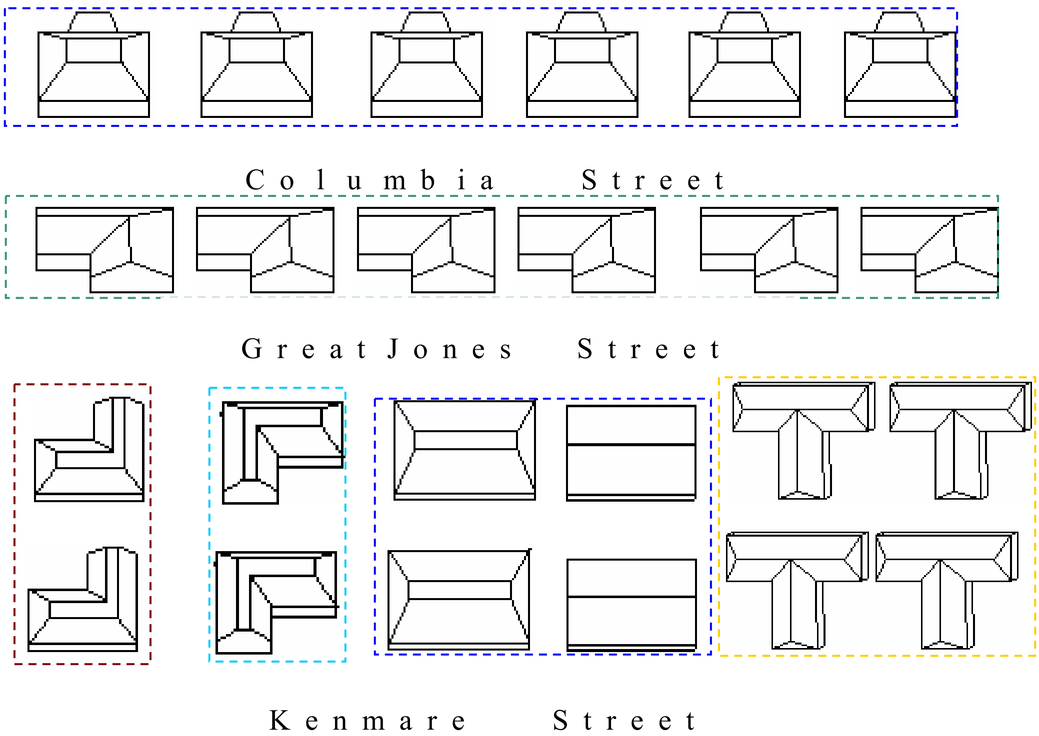


Figure 64:Principle of similarity

v. *Grouping by orientation* Buildings oriented along horizontal and vertical axes, or ones that are symmetric, are more often perceived as figures. Figure 65 shows such an example of an area of Munich city, where groups of buildings arranged and orientated in such a way that the whole arrangement appears in certain forms (i.e. circle, square etc. shown with different colors in the figure).

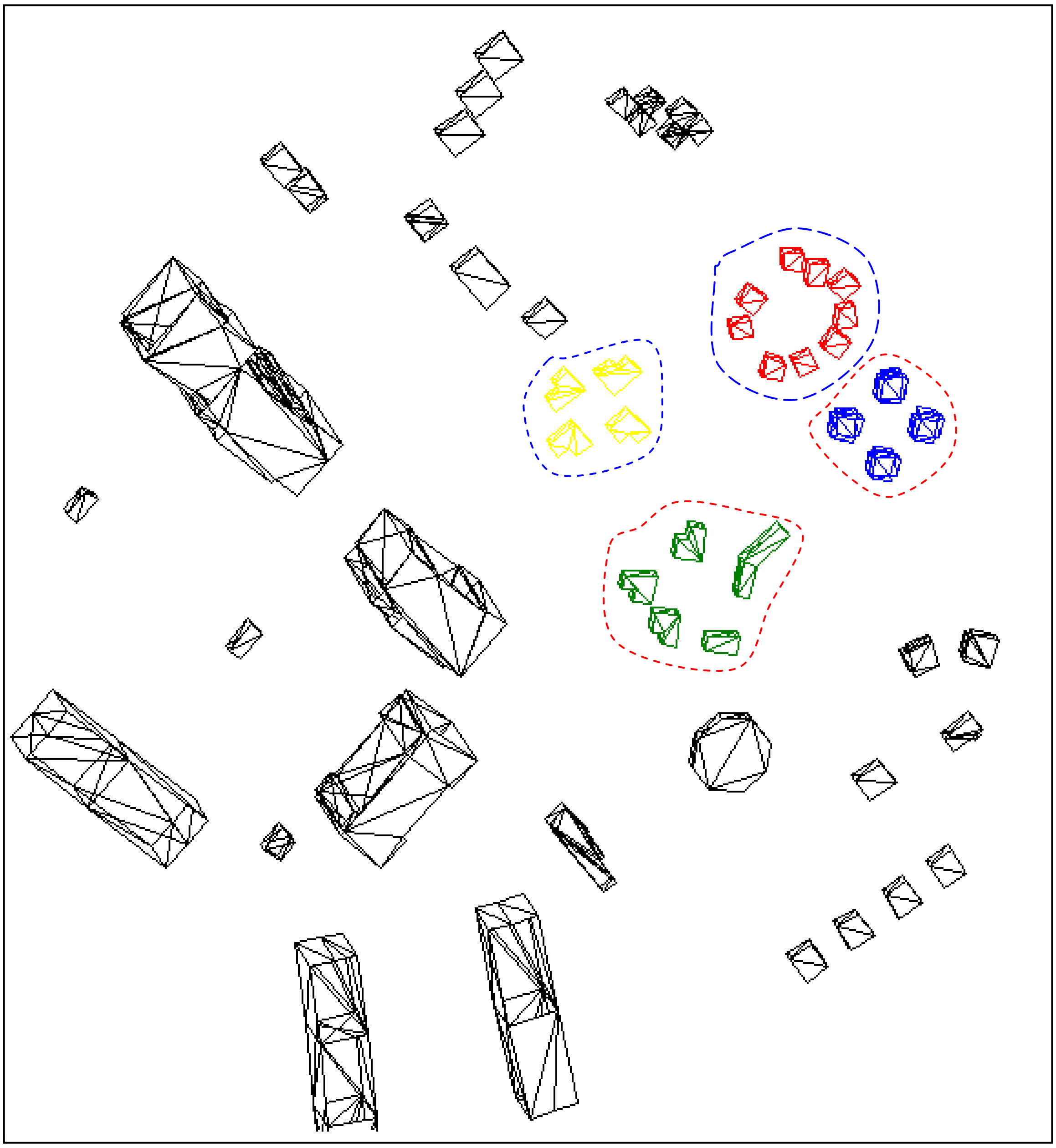


Figure 65: Principle of orientation

vi. *Grouping by co-linearity*: Structure recognition requires that buildings to be clustered should be within a desirable limit of co-linearity. The measure of the co-linear relationship is determined by summing up the deviations of the centers of the buildings from the line passing through the center of first building in the direction of other buildings as shown below in figure 66:

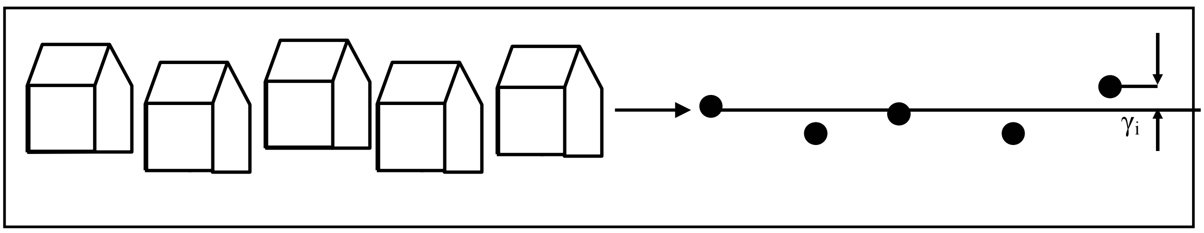


Figure 66: Principle of co-linearity

The measure of continuity $\boldsymbol{M_{col}}$ is given by

$$M_{col} = 1/n \sum_{i=0}^{n} \gamma_i$$

The smaller the value of M_{col} better is the co linearity. Most of the buildings lying along a roads are co-linear or within the desired limit and are clustered.

These processes play an important role in 3D generalization as will be evident shortly in next chapter where their effect on generalization in general and aggregation in particular are described in details.

The following table 9 gives a set of functions used to cluster buildings along both side of the road.

Sr. no	Function name	Usage
i.	GetRoadID	Get the Id of Road
ii.	DistFromRoad	Get distance from road to a building
iii.	CountRoads	Counts the total number of roads
iv.	IsEntityRoad	Returns the entity (Road or Building)
v.	UpdtRoadProximity	Compute proximity of all buildings to a nearest road
vi.	GetLeftOfRoad()	Get buildings left of a road
vii.	GetRightOfRoad	Get buildings right of a road
viii.	RoadBasedBldngClustering	Compute clustering with respect to all the roads
ix.	DisjointCluster	Get clusters using MSN tree algorithm

Table 9: Clustering functions

A structure recognition method has been studied here to find perceptually salient non-local structures. 3D buildings are grouped according to the gestalt principles of proximity, good continuation, and similarity based upon their shape, color, size or orientation. It completes the structure recognition of individual objects and objects in a group. This knowledge is utilized in the next chapter where rules are formed to govern generalization.

Chapter 7

3D aggregation based upon structure recognition

When a 3D scene, irrespective of its theme, is displayed on screen at a reduced scale, its contents do not decrease proportionally to the scale ratio applied. It, therefore, results in an increasing density of the contents due to small space available at smaller scale. That is where the generalization plays an important role. Generalization is the process of creating a legible map at a given scale from a more detailed geographical dataset. It is done in such a manner that the character or essence of the original features is retained at successively smaller scales. Though the purposes and benefits of generalization are manifold, it is indeed a complex decision-making process, which must be intelligently steered by goals and rules from the geographical application domain, such that the generalized representation conveys knowledge consistent with reality.

In recent past, lot of work has been done in 2D generalization (Beard 1991), (Weibel 1995), (Blanca 1995) and (Sarjakoski 1999), which defines a set of operations to be performed to achieve the desired result. Nevertheless, 3D generalization is altogether perceived differently. A given 3D urban area mostly consists of roads and buildings and these buildings are of different styles and features. Further, the city area may be viewed from different angles and at different heights. Therefore, generalization in general and aggregation in particular must deal with all these issues.

7.1 Operations of 3D generalization

Major operations of generalization include simplification, aggregation, typification, displacement, enhancement and symbolization. Simplification may be described as the elimination of unwanted details. Aggregation and typification reduce feature density at a given level of detail while maintaining the representative distribution pattern and visual impression of the original feature group. Displacement is implied to avoid conflicts among the nearest objects, whereas enhancement and symbolization aim at exaggerating important objects. Although all of these operations have been successfully implemented in 2D but only a few of them have been studied in 3D. In recent years, one of the important operations, 3D simplification has been introduced by (Forberg 2003) where they suggested an approach for the simplification of 3D building data. Here the idea of scale spaces applied in image analysis has been extended. Two scale spaces, a 3D version of mathematical morphology and the so-called 3D curvature space are applied separately, as both are suited for the simplification of different object structures. In a new approach introduced in (Forberg 2004a), the advantages of mathematical morphology and curvature space have been united in one process.

All of these operators use algorithm, which satisfy certain constraints to achieve the desire results so that generalized objects are displayed without, and conflict among them. These constraints require comprehensive structure recognition of the objects involved. As studied in previous chapters, structure recognition not only helps in derivation of measures from 3D individual buildings (i.e. area, volume, height, roof type, building type etc) and derivation of measures from two adjacent building (i.e. proximity) as well as buildings in group but also helps in derivation of constraints and the calculation of various associated parameters. These constraints are then become the source for the derivation of various rules for different generalization concepts. In this chapter, various kinds of constraints are found for 3D generalization, however, different rules are developed for aggregation as an example.

7.2 Constraints of 3D generalization

Constraints form the basis of rules**.** A constraint, as defined by (Ruas 1998), can be specified as something to maintain or something to avoid. Many constraints can be expressed in either way, such as "maintain a certain separation between two buildings", or "avoid overlapping of two buildings." Further, a constraint can be independent or contextual. Independent constraints consider only one object, e.g., a building's area must exceed a minimum size. Contextual constraints consider relations between objects, e.g., two buildings cannot occupy the same location. In general, we can classify constraints into the following categories:

- *Graphic constraints* arise from feature and symbol geometry. They specify basic size and proximity (i.e. area and distance) properties and are mainly dictated by graphic limits as well as the shapes and sizes of features. Examples for an individual feature include its minimal size, minimal width, minimal height, and minimal length. If multiple features are involved, graphic constraints define minimal separability and help to enforce proximity relations (Ruas 1998).
- *Topological constraints* ensure that basic topological relationships (connectivity, adjacency, containment) between features are maintained (Ruas 1998). For individual features self-intersecting lines and polyhedron boundaries should be avoided. When multiple features are involved, spatial transformations should not alter the topological relationships of the remaining features, even when these only indirectly represent the original features.
- *Structural constraints* define criteria that describe both spatial and semantic structure and inter-dependencies (Ruas 1998). Spatial structure on the level of individual features relates to shape (i.e. internal structure of feature) and its preservation (convexity/concavity of area).
- *Perceptual constraints* relate to aesthetics and complex perceptual aspects. They arise due to enforcement of visual balance (when typifying, aggregating etc.).

According to (Ruas 1998), all these constraints can also be classified into three categories:

- Macro level. Macro constraints characterize regional or overall map space; related to scope of semantic constraints;
- Meso level. Meso constraints characterize a neighborhood treated as a unit on a map, usually involving several to dozens of features of various types; related to scope of structural constraints; and
- Micro level. Micro constraints characterize a neighborhood defined by a feature, part of one or several that can be treated in isolation; related to scope of geometric and graphic constraints.

Constraints on roads and buildings have been studied here separately and is an extension of the constraints for roads and buildings in 2D as defined in (Ruas 1998):

7.2.1 Constraints on roads

Roads are the inherent part of the city and occupy considerable area of 3D city scene. When the scene is viewed at smaller scale, these roads try to mingle with each others and nearby building objects and therefore become source of constraints. Important road parameters, which lead to various constraints, are distance between a road and adjacent building (i.e. road shift), its orientation with near by objects, width of the road, density of roads, and its distance from nearby road. Table 10 shows the minimum values for them.

Parameter	Minimum (Threshold) value
Road shift from building	$\Delta S_{min} = 0.30\ mm$
Orientation	$\Phi_{min} = 5^o$
Road width	$w_{min} = 0.30mm$
Road shift from nearby road	$\Delta R_{min} = 0.30\ mm$
Density of roads	$\eta_{min} = 0.20\ mm^{-1}$

Table 10: Road parameters

Based on these parameters, followings are the constraints defined for roads at three levels.

(1) Micro constraints (road segments)

- Accuracy

  Roads should not move away to a large extent from their initial position. Small displacement, ΔS_{min}, may be permitted to avoid conflict with adjacent buildings.

- Orientation

  A road should preserve its initial main orientation, with respect to adjoining buildings and streets. A road can be broken into various parts with dominant orientations. The orientation of these parts is then compared with the generalized parts of the roads. The deviation should not be more than the allowed value, Φ_{min}.

- Shape

  A road should preserve its characteristic bends and its ups and downs, which depends on the predefined minimum dimensions. If a road has a bridge over or under it, it should be maintained. Again, Φ_{min} and width of the road will help in preserving the shape.

- Size

  Road width should be large enough (> ω_{min}) to avoid visual confusion.

- Functionality

  Roads leading to important buildings should be maintained. Isolated roads may be deleted if the adjacent buildings are removed.

- Semantics

  Relative importance of roads should be maintained.

- Look

  Texture, type of roads, and their colors should be maintained.

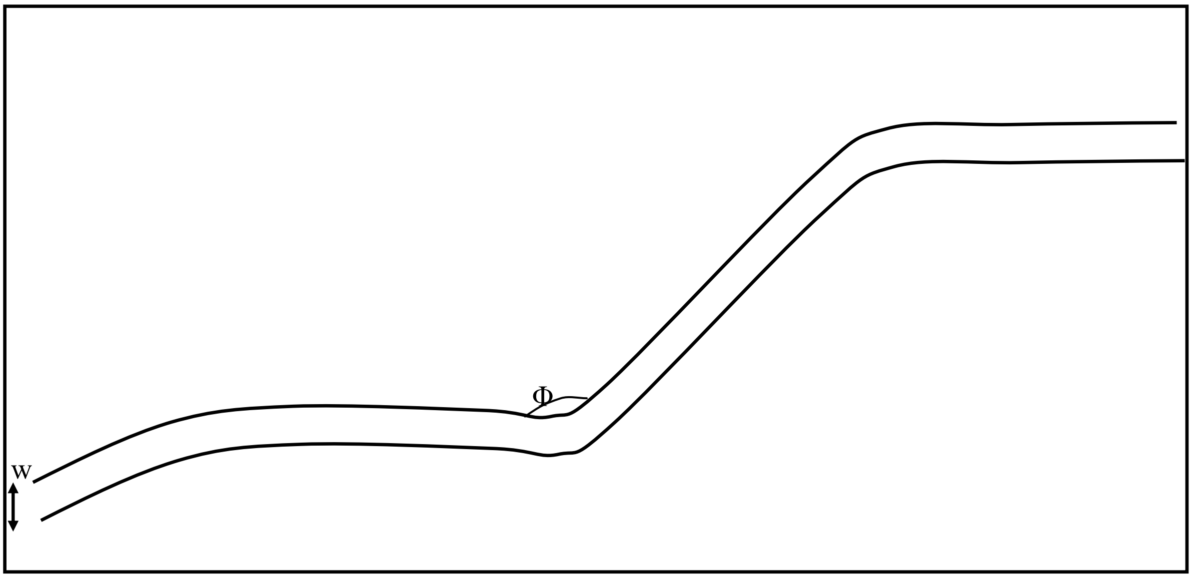


Figure 67: Road with different bends

2) **Meso constraints** on roads (routes, networks):

- Topology

  Existing connectivity, adjacency and inclusion relationships must be maintained if roads are maintained.

- Orientation

  Near and immediate neighboring roads ($\sim\Delta R_{min}$) should try to preserve their relative main orientation, if not deleted.

(3) **Macro constraints**

- Density

  Density of roads can be calculated as ρ = Total length in a given area / area

  Road networks should not have too high density ($< \eta_{min}$). Dense road networks can be simplified by removing isolated and small roads if the adjacent buildings are removed provided they are not leading to important buildings.

- Patterns

  The spatial arrangement of roads as shown in figure 68, may create distinctive patterns such as network, tree and radiating and should be maintained.

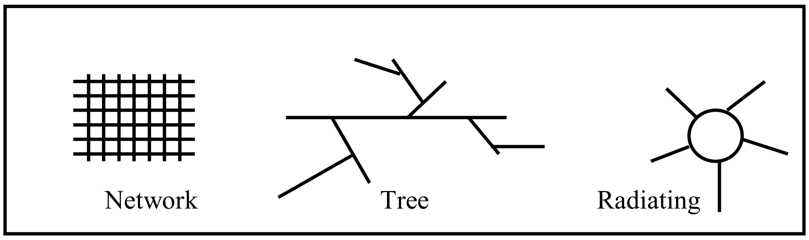


Figure 68: Different road patterns need to be preserved

7.2.2 Constraints on buildings

Important building parameters, which lead to constraints, are its length, width, height, angle, area and volume. Table 11 shows the minimum values for them.

Parameter	Minimum (or Threshold) size
Length, width	$l_{min} = 0.40mm$, $w_{min} = 0.40mm$
Height	$h_{min} = 0.45mm$
Angle	$\alpha_{min} = 5^o$
Area	$A_{min} = l_{min} * w_{min}$
Volume	$V_{min} = l_{min} * w_{min} * h_{min}$

Table 11: Minimum dimensions

(1) Micro constraints (polyhedrons)

- Accuracy

  Building should stay close to their initial position.

- Orientation

  A building should preserve its initial main orientation, with respect to adjoining buildings and streets, and absolutely if close neighboring features are not transformed.

- Shape

  A Building should preserve their main orthogonality. Buildings should maintain their characteristic elongation. If buildings have unusual shapes such as curves, these shapes should be maintained. Complex and aloof buildings should be exaggerated.

- Granularity

  Internal outline segments should keep their perceptible dimensions.

- Size

  Building size should exceed the minimum volume limit. Any side length should be large enough to avoid visual confusion.

- Functionality

  Important buildings should be maintained. Isolated buildings can be important buildings and may be used as landmarks.

- Look

  Texture, color and exterior outlook should be maintained.

(2) Meso constraints (building clusters within an area)

- Topology

  Existing connectivity, adjacent and inclusion relationships must be maintained if buildings are maintained (subject to aggregation).

- Orientation

  Neighboring buildings should try to preserve their relative main orientation.

- Proximity / repartition

  Smallest distance between two buildings must be greater than the perceptual threshold. The relative distance between two buildings should be maintained. Relative distances between a set of aggregated buildings should present the same clustering effect as is initially measured. Specific patterns and alignments (e.g. along curves) should be maintained. No specific distribution should be created if it does not exist initially (e.g. if buildings are not organized along a grid, they should not be after aggregation).

- Size distribution

  Size dilation should not disturb relative size order of buildings whenever their semantic meaning is the same. Two buildings, which have nearly the same sizes, can have equal final size

- Semantics

  Two buildings can be aggregated if they do not visually have contrasting semantic meanings. The distribution of different sub-classes of buildings within an area can change but proportion should try to be maintained. Some indication of semantic exceptions should be maintained (e.g. a commercial area within a neighborhood of detached houses).

(3) Macro constraints

- Quantity constraints

  Area and volume occupied by specific buildings should be maintained (for statistical study).

- Density distribution

  The ordering of density values among meso objects before and after aggregation should not vary much. Differentiation of densities should be still perceivable after aggregation.

7.2.3 Meso constraints on settlement blocks (road networks and building groups with road meshes)

- Topology

  Existing inclusion relationships between a building and a street partition must be maintained (i.e., buildings must not move across streets). Roof of the building or its balcony should not be extended to cover the adjacent road.

- Orientation

  Adjacent street and buildings should try to preserve their relative main orientation.

- Proximity / repartition

  The relative distance between a street and a building must be greater than separation distance and as close as it is initially.

- Functional dependency

  The functional dependency should be preserved after aggregation. Therefore, buildings should be removed/aggregated if their streets are eliminated, but generalization of street networks should allow for access to all buildings.

As it is clear from the study above, 3D has brought additional constraints due to extra dimension. These constraints will form the basis of new rules as compared to 2D. As emphasis is on 3D aggregation, therefore additional rules will be developed for it. Following paragraphs describe necessity of aggregation and will be followed by its rules.

7.3 Necessity of 3D aggregation

Aggregation has the task of grouping a selected set of like entities to form one entity by simplifying its representation over the original footprint. Though 2D aggregation, which deals with polygons, has been widely studied in past, aggregation of 3D objects (polyhedrons) has not drawn considerable attention until recently. Here the focus will be 3D aggregation for a city model. While aggregating the 3D city, several important issues need to be considered. Apart from the existing rules and constraints derived from the 2D aggregation, additional 3D rules and constraints should be taken into account. In case of 2D, we have only one ground view and aggregation rules can be confined to that view. But in case of 3D, where detailed 3D city models should also be accessible to investors, builders, urban managers, and tourists for a variety of purposes, different views of the same area need to be studied and consequently suitable aggregation algorithm should be developed, which takes into account the visual importance of specific features during aggregation. A 3D view not only has ground view but additional perspective view as well. In 2D ground view, most of constrains results from the measures of its ground plan, proximity, area etc. However 3D views not only include these constraints but the additional constraints, as studied above, resulting from its different perspective view, height of the buildings, its historical importance, location and many others.

Aggregation may be performed for a number of reasons:

a. When the density of buildings within an area is high, resulting in conflicts such as overlapping symbols, the buildings should be aggregated into a simplified representation. Aggregation may be performed with a view to reduce the overall impression density.

b. When one or more buildings are too small to be represented individually, the buildings may be enlarged. This enlargement of buildings may lead to an impression of over-occupation in an area. To counter this, if the buildings are adjacent to or in close proximity of one another, aggregating them into a composite representation may be performed. Generally, it is attempted to maintain the structure and distribution of the group following the operation.

c. When it is wished to enhance or exaggerate the impression of density distribution of information across a 3D scene, either for reasons of enhancing communication or to maintain consistency in representation amongst different parts of it, aggregation may be performed. This operation needs to be controlled at a strategic level, where different gestalt issues must be considered.

7.4 Rules of 3D aggregation

The aggregation is governed by a set of rules formed on the basis of structure recognition studied in the previous chapters, which form the core of the aggregation algorithm. These rules are divided into two categories - stiff and elastic rules and will be specific to roads and buildings.

7.4.1 Stiff rules

Stiff rules, as the name implied, must be applied strictly to pursue aggregation. These rules can be divided into four sets of rules as follow:

- Linkage rules
- Semantic rules
- Structural rules
- Orientational rules
- Contextual rules.

i. *Linkage rules* define the spatial relations between buildings that must exist among them for their aggregation (Ruas 2001). These rules include topological and metric relationships such as:

- Proximity - The buildings to be aggregated must be disjoint but within a certain distance of each other for their aggregation.
- Alignment - The buildings to be aggregated should be aligned or their alignment should not differ much and the difference must be below the permissible limits.
- Side of road: The buildings to be aggregated should be on the same side of the road.
- Adjacency - The buildings to be aggregated must have their adjacent faces close.
- Touching - The buildings to be aggregated must have their common adjacent face touching together.
- Angle: The angle between the buildings to be aggregated should be less than the threshold value.
- Size of the buildings to be aggregated should be greater than a given threshold value.
- Height - The buildings to be aggregated should have minimum height difference. However, if the two buildings vary much in height, it may be possible, depending on the requirement, that both are simply united.
- Roof Type - The buildings to be aggregated should have similar types of roof i.e. planar or gable.
- Building Type - The buildings to be aggregated should be of similar types i.e. simple or complex.

ii. *Semantic rules* define the semantic relationships that must exist for aggregation. Semantic rules include relationships such as:

- The buildings to be aggregated should belong to same class (e.g. public or private).
- The buildings to be aggregated should keep their association intact even if they lie both sides of a road.
- The buildings should be aggregated if they belong to same owner.
- The buildings should be aggregated if they all are old or new.

iii. *Structural* rules apply to a group of buildings forming a common geographic or perceptual structure. These groups are so organized such that the human visual system spontaneously recognizes them as a whole without any semantic knowledge.

- Set of buildings, which are in close proximity forming a group, should keep their closeness after their aggregation to enhance proximity perception.
- Set of buildings, which are located on a continued line/curve, should keep their perception of continuation after aggregation.
- Group of buildings, which are co-linear, should preserve co- linearity after aggregation.
- Group of buildings forming a closed structure, should preserve their perception of closeness after aggregation.
- Group of buildings, which are orientated at same angle and are in proximity, should preserve perception of orientation after aggregation.
- Group of similar buildings should keep the similar perception after aggregation.

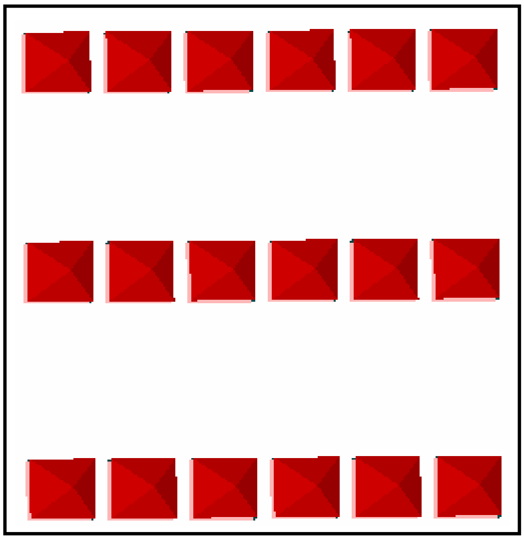
iv. *Orientational rules* define the historical or local importance of the building. A city consists of few buildings that are relatively more important and unique (e.g. uniqueness in comparison to nearby buildings with salient size, height, texture etc.) then the rest of the buildings. For example, either a building is very tall such as a tower, a huge structure such as a palace, an architectural monument and reflects city identity**.** These building should remain as unique landmarks. For example, the famous church (Frauenkirche) in Munich, may immediately make the viewer, who has seen the city at least once, understand that it is Munich city**.** Therefore the following rules can be defined:

- Buildings, which are of great importance, should have their identities preserved to the greatest extent possible**.** Therefore, two buildings having historical importance should not be aggregated even if others rules are satisfied.
- If only one of the buildings is important, even then aggregation should not be done. Instead, the important building should be exaggerated. The same rule applies to other local important buildings like TV towers, theatres, schools etc.

v. *Contextual rules* define the sensitivity of aggregation caused by 3D using contexts. Different features of the buildings should be preserved for different occasions. In the manual era and 2D, it was done by an experienced cartographer with an understanding of the map context and the semantic meaning. In developing automated techniques, the challenge is to embed this contextual and semantic knowledge into the software. Contextual rules include followings perspective cases as far as aggregation is concerned:

(a) Top view – It highlights the topside from a viewpoint cantered over the topside as shown in figure 69. It has a very serious drawback in the sense that it is relatively difficult to understand because it does not convey a sense of depth; hence, the shape of many surfaces appears ambiguous. It is hardly different from the 2D view and therefore corresponds to an unusual view in 3D environment.

(b) Oblique view - An oblique view is not orthogonal to or cantered over any side. When the view angle is properly selected, it presents a good overall look of an object, as it would be seen in real life. It conveys a good depth cue and is easier to understand than top view as shown in figure 70. Depending upon the height, from where the viewer is interested in viewing the generalized scene, different features of the buildings play an important role and should be a part of aggregation. Under perspective view, roof of the buildings should be major criteria for aggregation as it helps to maintain the visual balance. It may be desired that even with other building attributes varying slightly from the threshold values, but with same roof types, should be selected for aggregation.



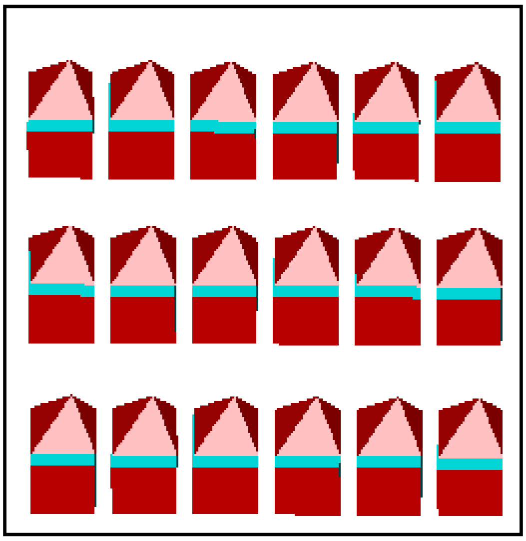


Figure 69: Top view **Figure 70: Oblique view**

Two representative oblique views depending on different viewing heights can be differentiated from each other: the ground perspective view and the bird's eye view. When the viewer is standing on the ground, walking or driving, the resulting view is a ground perspective view in which features such as walls, windows, and doors are visible to the viewer while roof type may be of least importance. In this case, buildings with different roofs can be aggregated as long as they share sufficient common attributes satisfying other aggregation rules. On the other hand, the roof type is a decisive eye-catching feature in the bird's eye view in which the viewing height lies in the air, that is to say, above most buildings as shown in figure 71.

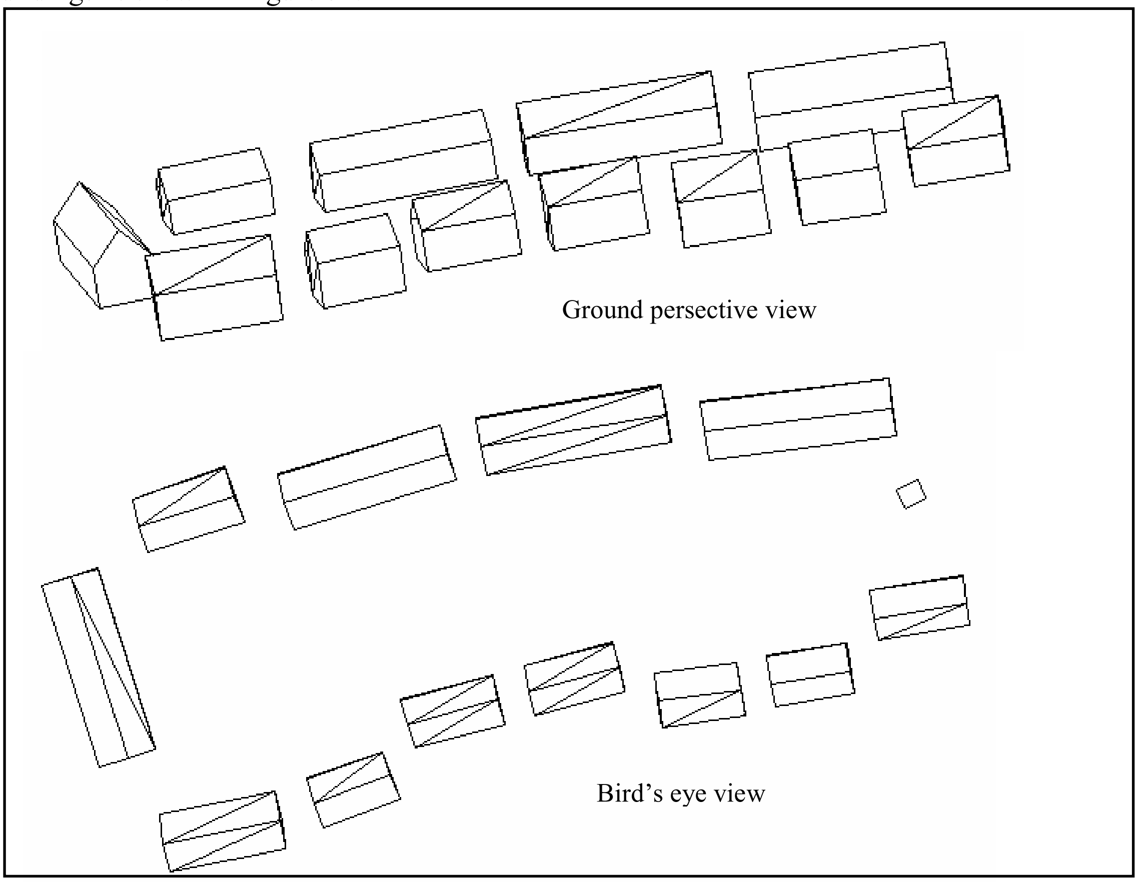


Figure 71: Example of ground perspective view and bird's eye view

In a bird's eye view, roof types play an important role for the recognition of buildings whereas the lower portion of the buildings is less conspicuous. Therefore, particular care should be taken while aggregating buildings having different roof types.

When the buildings are viewed from a balloon or a helicopter flying over it, the height of the viewer increases and consequently the height of the roof appear to decrease and is almost half the original height but the buildings are still distant apart as shown in figure 72. In fact, the higher the viewer is located, the larger is the flattening effect of the roof, gable for example. Figure 72 and 73 reveals the impression of a building block based on the view from a flying balloon.

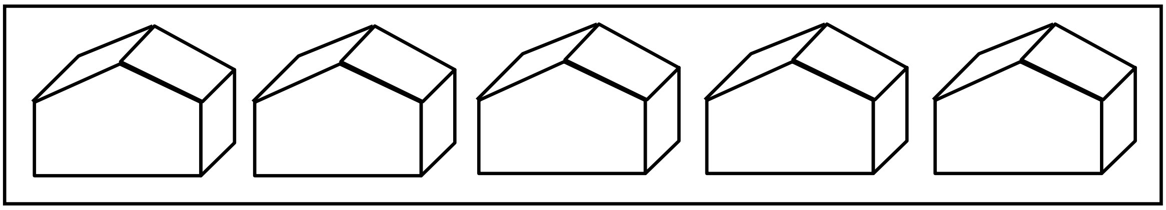


Figure 72: perspective view from a flying balloon or helicopter

Figure 73: View from a flying balloon

One of the difficulties arising in 3D generalization is due to the contextual rules. These rules deal with different views when the viewing parameters can be anyway controlled by the users. It demands that aggregation should be done on the fly in real time. It is indeed a challenging problem and should be looked into in future.

7.4.2 Elastic rules

Elastic rules are rules, which may or may not be applied. The importance and applicability of an elastic rule may depend upon circumstance and vary from case to case. Elastic rules can be derived based on the context-sensitive importance of the following features of the buildings: texture, exterior outlook, color and are discussed below:

The texture of the buildings in general and of roofs in particular plays a significant role to achieve a photorealism of a generalized 3D city model. It becomes even more prominent when aggregating historical buildings because their original look should be reiterated. In general, buildings have a number of character-defining aspects, which include the windows and the decorative stonework as shown in figure 74, but certainly, the roof and roof features contribute more to the overall visual character. The roof is not only highly visible, it may have elaborate stone dormers and it may also have decorative metalwork and slate work. The red and black slates of differing sizes and shapes are laid in patterns that extend around the roof of a large and freestanding building. Any changes to this patterned slate work, or to the other roofing details would damage the visual character of the building.

Figure 74: Texture of a building

Figure 75: Characteristic roof of a building

Color plays such an important role in emphasizing the unique characteristics of buildings. Building elements such as windows, doors, walls, columns, roofs, domes are all painted in various colors, which create a vivid image. Even from a distance, colored images may well strike tourists as potent and memorable, thus creating a lasting mental picture, which may lure tourists to revisit the place. For instance, the historic Dutch and British buildings (the Stadthuys Building and the Christ Church) in the old town of Malacca were all painted in red. For the same reason, most historic buildings in the city of Edinburgh, Scotland appear in grey. Both examples present a clear and lucid image of the area as well as the entire city. The unique red buildings in Malacca and grey buildings in Edinburgh may well be highlighted and promoted as tourism products. So while aggregating two buildings with different colors of roofs and walls, care must be taken to preserve the original colors, at least of dominant building.

Exterior outlook is another important aspect of a building character. A building roof may consist of dormers and chimneys. The roof in figure 75, for example, is important to the visual character because its steepness makes it highly visible, and its prominence is reinforced by the patterned tinwork, the six dormers, and the two chimneys. Changes to the roof or its features, while doing generalization, such as removal or alterations to the dormers, for example, would certainly change the character of this building. This does not discount the importance of its other aspects, such as the porch, the windows, the brickwork, or its setting; but the roof is clearly crucial to understanding the overall visual character of this building as seen from a distance. A projecting porch or balcony can be very important to the overall visual character of almost any building and to the surrounding in which it is located. Despite the size of this building (3-1/2 stories of figure 76), its distinctive roofline profile and the importance of the very large window openings, the lacy wraparound iron balcony is singularly important to the visual character of this building. It would seriously affect the character to remove the balcony, to enclose it. Therefore, the external outlook of the building should be preserved to the maximum possible extent.

Figure 76: Characteristic exterior outlook of a building

7.5 Divide and conquer algorithm for aggregation

Based upon the rules studied above, aggregating a given area of the city as a whole may still lead to confusing results. The most serious drawback of this approach is the lack of visual balance. For example, figure 77a, where groups of buildings are situated on either side of the road. After aggregation, it may happen that one of the buildings along the road may get into aggregation with another building lying nearby (figure 77b) but still away from the road. So a new strategy has been developed and called divide and conquers aggregation. Here only those building are chosen which lie close to a given road and aggregation is applied as shown in figure 77c. This process is repeated until all the roads are scanned.

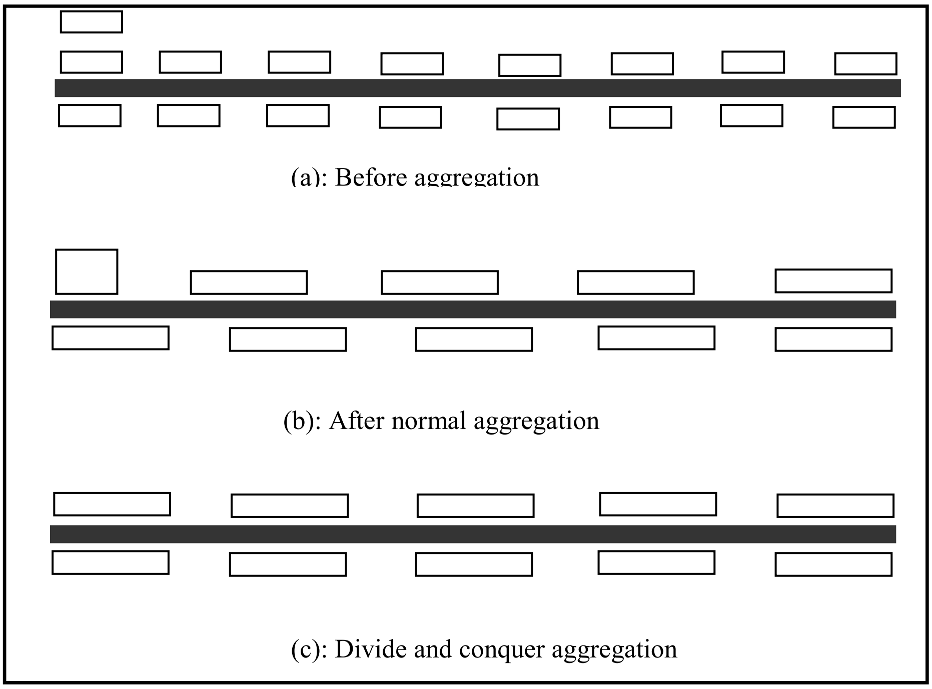


Figure 77 : Divide and conquer algorithm

The simple algorithm takes the following form and is basically a nested set of If..Else statements.

If $\boldsymbol{d}(o_i,o_j) < \Delta d_{\mathbf{min}}$ then

If $\boldsymbol{h}(o_i,o_j) < \Delta h_{\mathbf{min}}$ then

If ***roof_type₁*** = ***roof_type₂*** then

If $\Delta\mathbf{A}(o_i,o_j) < \mathbf{A}_{\mathbf{min}}$ then **aggregate**

…

Else if **Area**$_i$ is small and Object is ***important*** then **exaggerate**

…

Else **unchanged**

Once the buildings along the roads are scanned, then the remaining buildings, if any, can be taken care of based upon above rules and constraints. Nevertheless, this situation does not arise at all.

A comprehensive flowchart of the above algorithm is shown in figure 78.

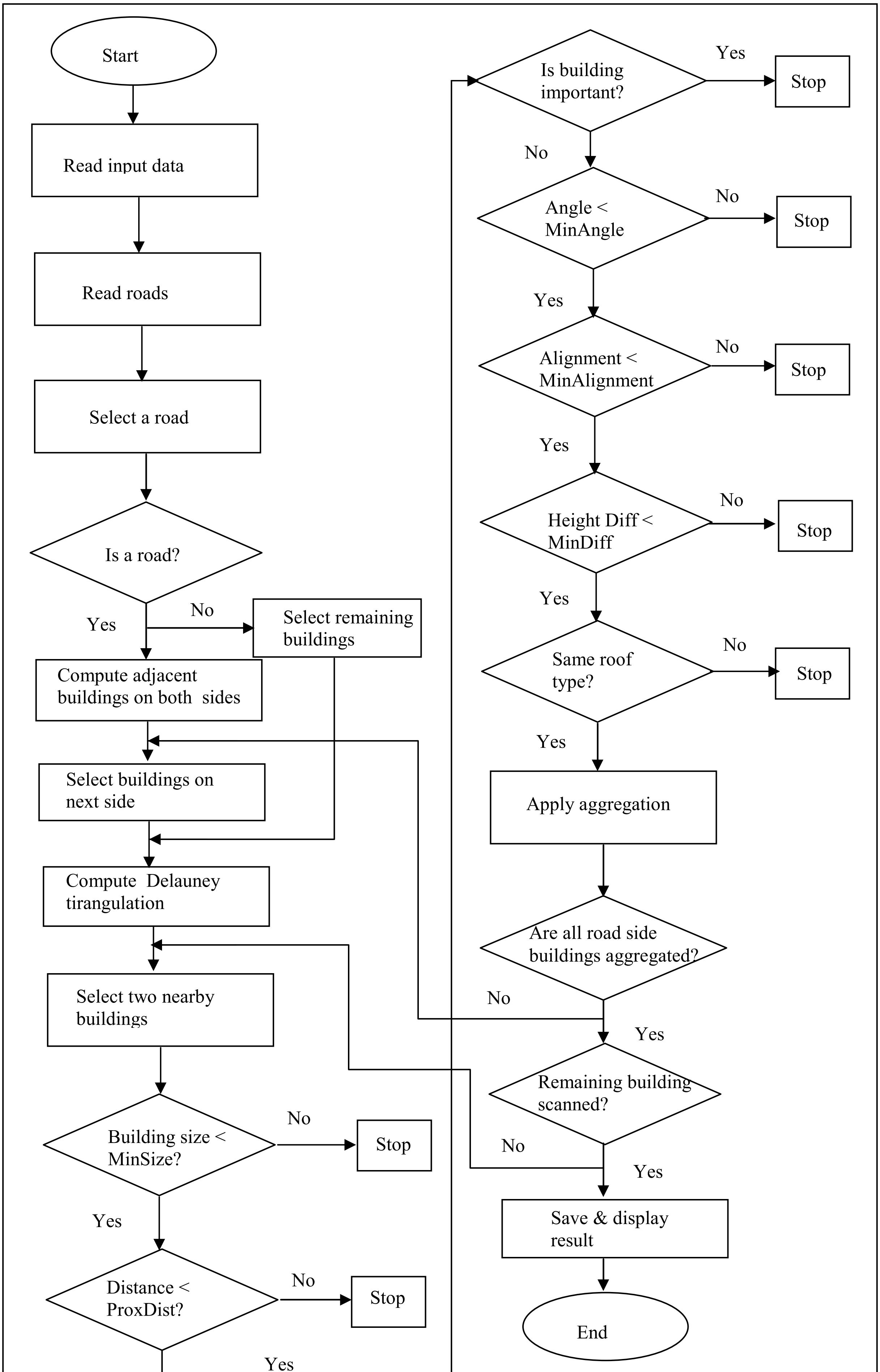


Figure 78: Flowchart of aggregation algorithm

7.6 Implementation and results

Above algorithm is implemented using Object Oriented approach under Visual C++ environment and on MS Window platform.

A test data consisting of few buildings of different roof styles and roads is constructed and is used to test this algorithm as shown in figure 79.

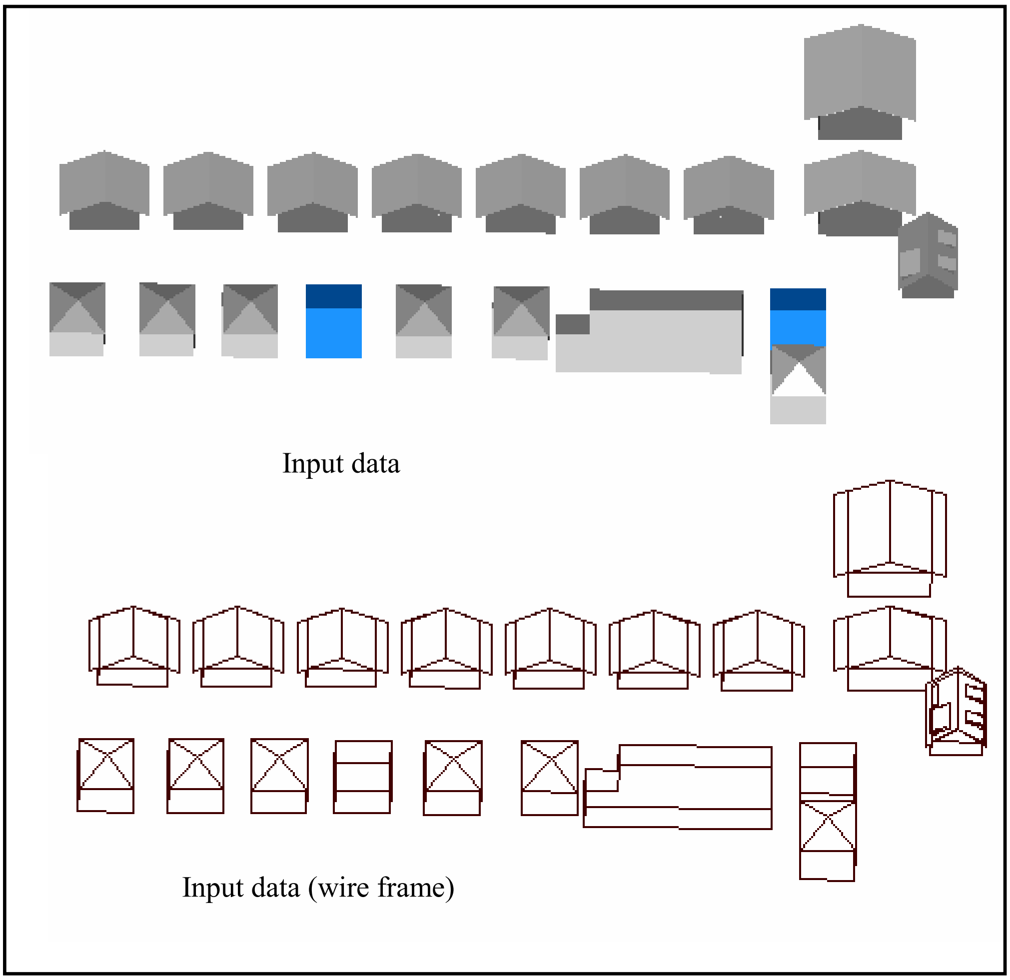


Figure 79: Original input .sat file

It consists of simple buildings that are located along roads. To maintain the visual balance, buildings are clustered along the road as discussed in chapter six. Two buildings are taken from a cluster at a time and a set of rules is applied to them. If they pass through these rules successfully, then they are aggregated, otherwise they are ignored. This process continues until all the buildings are processed. The aggregated as well as non-aggregated buildings are displayed in one color as shown in figure 80. As seen in the figure, All the buildings lying along the road (at the top of the figure) are of same type and are selected for aggregation. Another two buildings (with blue color) are aggregated together with the same reasons. However, there are some other buildings, which don't find any similar building in their vicinity and are not selected for aggregation (having cyan and grey colors). It may be necessary to apply other generalization operation(s) to them.

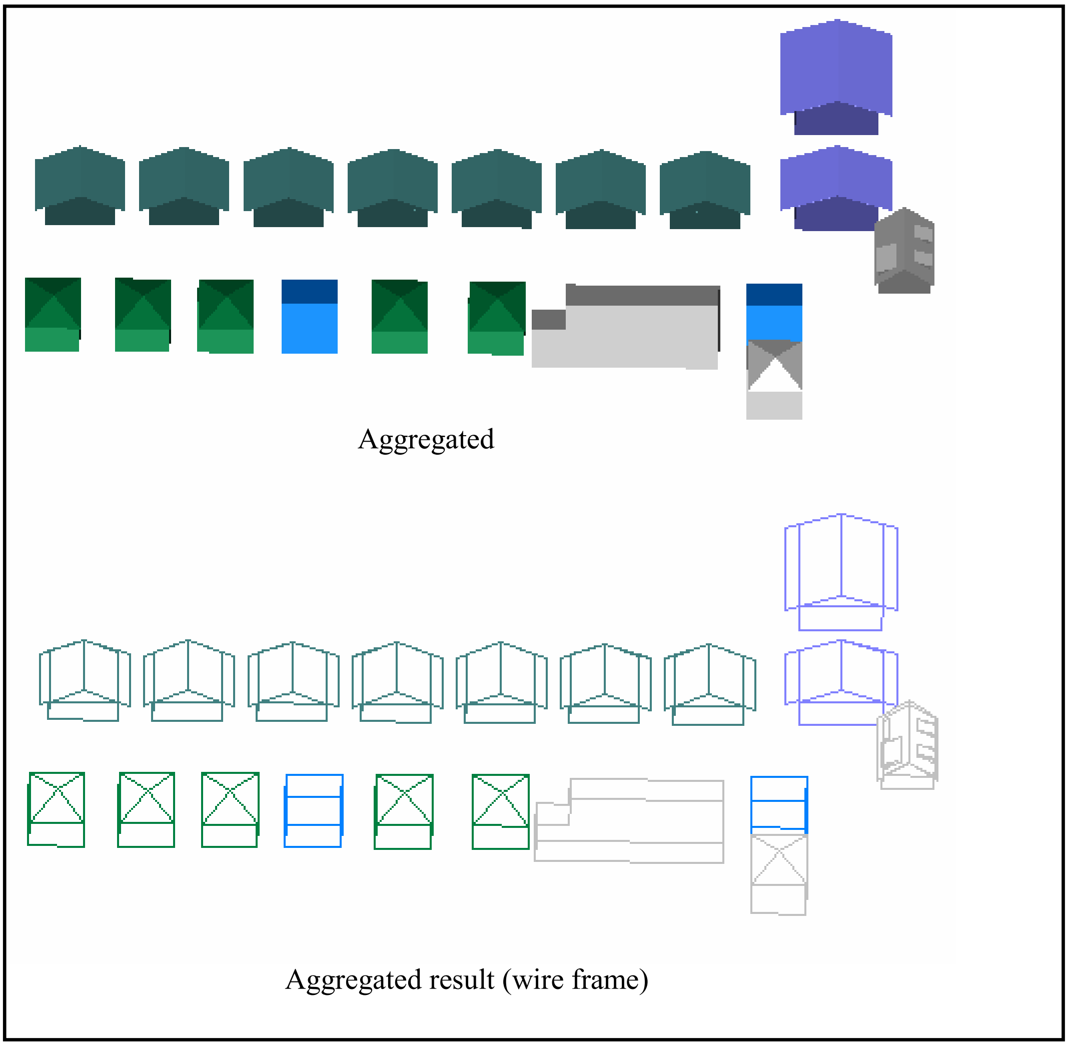


Figure 80: Building identified to be aggregated are shown with same colors

Though the data set is relatively simple but it shows courageous results. Buildings, which are selected for aggregation, have not been replaced with new buildings. Due to limitations of the software, these buildings could not be created. Although only small sets of simple buildings are taken however, the verification of the algorithms is done underlining the importance of structure recognition.

In this chapter, various rules are developed based upon different constrains. These constraints resulting form the structure recognition study. An algorithm for aggregation of 3D objects is developed and implemented. This algorithm is then tested on a test data of 3D buildings.

Chapter 8

Final discussion and conclusion

8.1 Conclusion

The prime aim of this thesis, as the title suggest, is to study structure recognition of 3D settlements and their use in 3D generalization. The focus was on the quantification of spatial characteristics at micro-, meso-, and macro-level. The research findings from 2D structure recognition had been extended to 3D, which led to the identification & definition of some new parameters Therefore; a number of new parameters were identified and defined. The results of structure recognition were then applied to test the functionality of a 3D aggregation algorithm guided by constraints and rules.

In the literature, review (chapter 1) a case was made to study the importance of the structure recognition towards generalization. Despite all the research that was reviewed, it is found that structure recognition was not given its due importance in 2D. It was noted that though the majority of the research undertaken in 2D generalization has mentioned structure recognition as an important issue but not much work has been reported there about it. It may be due to the fact is that most of the efforts have been done in realizing various generalization algorithms and operators.

Chapter 2 outlines the *state of the art* for structure recognition. Various methods have been reviewed for structure recognition here. It is found that existing research of structure recognition has been largely restricted to 2D. These approaches have their own drawbacks for their uses in 3D and have been adequately justified.

Chapter 3 deals with a hierarchical study on structure description that was divided into micro, meso and macro levels. In micro-level, which deals with the individual objects, the study of its various parameters such as, positional parameters, form parameters, orthogonalities, roof type and general shape of the ground-plane has been emphasized. In meso level, spatial relationships among individuals and neighborhood objects are studied. This level is the most important part of the study as most of the rules and constraints are later derived for generalization in general and aggregation in particular. Some of the parameters studied here are proximity, height difference, angle, alignment, size contrast, aloof but important building, different roof styles and different building types. In macro-level, clusters of objects having similar properties such as settlement/building blocks are considered. Its detection is based on human perception, especially the visual grouping behaviors. The clusters are an important aspect in understanding images, maps and 3D scenes and therefore are comprehensively studied. Main emphasis is on

- Shape and size regularity
- Regularity of roof structure and surface texture
- Adjacency to other structures
- Unique, deterministic features
- Relative distributional density

ANN technique was applied to recognize different buildings (chapter 5). Though no work have been reported in the past for building recognition using this AI technique, inspiration was taken from literature survey on pattern recognition, which employs this technique very effectively in face and hand written character recognition. Though face and character recognition is a 2D recognition but it has been efficiently and successfully extended to 3D.

Chapter 6 studies the structure recognition of group of buildings forming a cluster. Principles of perceptual grouping are studied here and are required to maintain the visual balance while applying generalization. Most important principles given emphasize are

- Grouped by continuity
- Grouping by similarity
- Grouping by continuation
- Grouping by parallelism
- Grouping by co-linearity
- Grouping by proximity

Since it a city model, most of the buildings are situated along the roads. When these buildings are first grouped based upon their alignment to a road, these perceptual principles are implicitly applied when other structure recognition based rules are executed on them. Therefore, aggregation algorithm developed in this study; first separates all the buildings into various groups based upon their road adjacency and then other rules are applied to them.

With this study, it has become evident that it is now easier to understand the topology and spatial relationships among the main city objects. This study has resulted in forming the strong base for determining the various rules and constraints for generalization. Aggregation, one of the most important components of the generalization was studied (chapter 7) here. Various levels of constraints were studied for roads, buildings and road & buildings in proximity. Based upon these constraints, aggregation rules were formulated and algorithm was developed and implemented. Two types of rules were identified; stiff and elastic rules. Stiff rules are those rules, which must be applied while doing aggregation as they are strictly based on structure recognition. The importance and applicability of an elastic rule, on the other hand, may depend upon the circumstance and may vary from case to case. These elastic rules may be derived based on the context-sensitive importance features of the buildings such as texture, exterior outlook and color.

Another important aspect, which was studied here, is based upon c*ontextual rules*, which define the sensitivity of 3D aggregation towards different contextual views. Different features of the buildings should be preserved for different occasions and therefore this contextual and semantic knowledge has to be embedded into the algorithm. Two views are considered viz. top view and oblique view and their resulting effects are considered.

This study is among the first attempts which concentrated on 3D generalization, based upon exhaustive structure recognition research, and where different contextual views are considered.

8.2 Problems encountered

While pursuing research, problems are due to come. As not much research work was done on 3D structure recognition and aggregation in the past, lot of problems encountered during the research. One of the biggest problems faced during the whole research had been the real city data in the required format. As the whole development was based around Visual C++ and ACIS geometric modeler, which uses its own file (*sat*) format. It was not possible to get the required data despite enquiry made to various related sources. Finally, it was decided to develop a conversion utility, which converts *vrml* format to *sat* format. Fortunately, the building data was available for BONN city in *vrml* format. After converting the data into the required format, it was used for further research.

Though vast literature was available on generalization, very little was reported on 3D generalization. Structure recognition had been hardly studied in context to generalization, though its importance was mentioned for generalization but no research paper was found which could give a deep insight into it.

8.3 Future work

There are several aspects of the study, which raise important considerations for future research in the area of 3D structure recognition and generalization.

Various types of buildings, simple as well complex, were identified but still there is ample scope for different varieties, particularly in European cities, to be explored and consequently new relations among them may be found. Recently, new architectural buildings are coming up having entirely different shapes (viz. new build up area in La défense PARIS). Exploration of them will certainly require additional rules and constraints.

Due to the additional dimension, different views of a given 3D scene have come into considerations. Various constraints have been introduced and rules were added to aggregation algorithm. Further study may focus on constraints and rules for other generalization aspects such as displacement, typification, simplification, amalgamation, and exaggeration etc. and consequently their algorithms may be incorporated with these changes. Study on structure recognition made here will serve as the basis for addition rules and constraints.

AI techniques, ANN has been applied successfully to structure recognition and has shown excellent results. Most important advantage of this technique is that they are capable to predict good results even if the input features are transformed, scaled, rotated, incomplete, or little different from the already learned one. This approach may be continued for rest of the generalization algorithms.

Bibliography

Ahuja, R. K., Magnanti, T.L., and Orlin, J.B (1993). Network Flows: Theory, Algorithms and

Applications, Prentice-Hall, Englewood Cliffs, N. J.

Anders, C. (2003). A Hierarchical Graph-Clustering Approach to find Groups of Objects. ICA Commission on Map Generalization, Fifth Workshop on Progress in Automated Map Generalization, PARIS.

Andina, D. (2001). Artificial Neural Networks, Universidad de Guanajuato.

Andrea, K. H., and Sester, M. (1997). "Methods of data base interpretation-Applied to model generalization from large to medium scale."

Bader, M. (2001). Energy Minimization Methods for Feature Displacement in Map Generalization. Zurich, University of Zurich.

Bader, M. a. B., M. (2001). Cartographic Displacement in Generalization: Introducing Elastic Beams. 4th ICA Workshop on Progress in Automated Map Generalization,, Beijing.

Bard, S. (2003). Evaluation of generalisation quality. ICA map generalization, PARIS.

Bassett, R. A. (2003). Machine assisted grading of rare collectibles through the COINS framework. School of comp. sc. and info. systems. New York, Pace University.

Beard, K., and Mackness, W. (1990). Development of an interface for user interaction in rule based map generalisation. Proceding GIS/LIS '90, CA.

Beard, K. (1991). Generalization operations and supporting Structures. Proc. Auto-carto, Baltimore.

Bertin, J. (1983). The semiology of graphics: Diagrams, networks, maps. Madison, USA, Univ. Wisconsin Press, Madison.

Biederman, I. (1987). "Recognition by components: A theory of human image understanding." Psychol. Review 94: 115-147.

Bjørke, J. T. (1999). Map Generalization: A Fuzzy Logic Approach. ICA workshop on Map Generalisation, Ottawa.

Blanca, B. (1995). Digital Map Generalization in practice. workshop on progress in Automated map generalization, Barcelona.

Blanca, B., and Maria, P. (1999). Map Generalization to Obtain the Topographic Map of Catalonia 1:10,000. ICA workshop on Map Generalisation, Ottawa.

Bobrich, J. (2001). Cartographic Map Generalization in Urban Districts. Proceedings of the GIS Research UK, 9th Annual Conference, UK.

Boris, A., Alexei, V.F., and Sanner, M. (1997). "Recognition of protein structure on coarse lattices with residue–residue energy functions." Protein Engineering 10(10): 1123-1130.

Brachmann, R. J. (1997). "On the Epistemological Status of Semantic Networks." In: Associative Networks: 191-215.

Brassard, L. (2000). The perception of the image World. Burnaby, Canada, Simon Fraser University, Canada.

Brassel, K. E. a. W., R. (1988). "A Review and Framework of Automated Map Generalization." Int. Journal of Geographical Information Systems 2(3): 229-244.

Brenner, C. (2000). "Towards fully automatic generation of city models." ISPRS 23.

Brenner, C., Sester, M. (2003). Continuous Generalization for Small Mobile Displays,. Int. Conference on Next Generation Geospatial Information, Boston, MA.

Burghardt, D., and Meier, S. (1997). Cartographic Displacement Using the Snakes Concept. Semantic Modeling for the Acquisition of Topographic Information from Images and Maps, Birkhäuser Verlag.

Burghardt, D., and Cecconi, A. (2003). Mesh Simplification for Building Selection. ICA map generalisation, PARIS.

Buttenfield, B. P. (1995). OO map generalization: modelling and cartographic considerations, Taylor & Francis.

Byron Nakos, B., and Mitropoulos, V. (2003). Local length ratio as a measure of critical points detection for line simplification. ICA, Map Generalisation, IGN. PARIS.

Choi, Y. H., and Li, Z. (2001). Correlation of Generalisation Effects with Thematic Attributes of Cartographic Objects. 4th ICA Workshop on Progress in Automated Map Generalization, Beijing.

Christophe, S., A. Ruas, (2002). Detecting building alignments for generalisation purposes. Sympsium on eospatial theory, processing and applications, Ottawa.

DeLucia, A., and Black, T. (1987). A comprehensive approach to automatic feature generalization. Proceedings of the 13th International Cartographic Conference, Morelia, Mexico.

Ding, J., Xiaohuan, Y., and Wang, N. (2001). "From 1D to ND fractal in Automated cartographic generalization." Journal of Geographical sciences 11: 86-90.

Djouadi, Y. (1994). Cartage: a cartographic layout system based on genetic algorithms. 5th European Conf. on Geographical Information System.

Duchene, C. (2003). Coordinative agents for automated generalisation of rural areas. 5th Workshop on "Progress in Automated Map Generalization", IGN Paris.

Dudani, S. A., Breeding, K. J., and McGhee, R. B., (1997). "Aircraft Identification by Moment Invariants." IEEE Transactions on Computers C-26(1): 39-46.

Edwardes, A., and Mackaness, W. (1999). Modelling Knowledge For Automated Generalisation of Categorical Maps - A Constraint Based Approach. ICA workshop on Map Generalisation, Ottawa.

Edwardes, A., Burghardt, D., Bobzien, M., Harrie, L., Lehto, L., Reichenbacher, T., Sester, M. and Weibel, R. (2003). Map Generalisation Technology: Adressing the Need for a common Rerearch Platform. Proceedings of 21st International Cartographic Conference, Durban/South Africa.

Forberg, A., Lal, J. (2002). Generalization of Three-Dimensional (3D) settlement structures based on scale-spaces and structure recognition. Munich, TUM and BW, Munich.

Forberg, A., and Mayer, H. (2003). Squaring and Scale-Space Based Generalization of 3D Building Data. ICA Map Generalisation, PARIS.

Forberg, A. (2004a). "Generalization of 3D building data based on a scale-space approach." The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXV(Part B): 194-199.

Forberg, A. (2004b). Simplification of 3D building data. ICA Workshop on Generalisation and Multiple Representation, Leicester, UK.

Förstner, W. (1999). 3D-City Models: Automatic and Semiautomatic Acquisition Methods. Photogrammetrische Woche. Stuttgart.

Galanda, M. (2001). Optimization techniques for polygon generalization. 4th ICA Workshop on Progress in Automated Map Generalization, Beijing.

Galanda, M. (2003). Modelling constraints for polygon generalization. ICA Map Generalisation, PARIS.

Garriga, M., and Baldwin, G. (2001). Generalization of Multiple Scale Maps from a Single Master Database. 4th ICA Workshop on Progress in Automated Map Generalization, Beijing.

Geoffrey, H. D. (1999). Visualizing Map Generalization Solutions on the World Wide Web. ICA Workshop on Map Generalisation, Ottawa.

Goldberg, D. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley.

Granlund, G. H. (1972). "Fourier Pre-processing for Hand Print Character Recognition." IEEE Transactions on Computers C-21: 195-201.

Haeberling, C. (1999). Symbolization in topographic 3D maps – conceptual aspects for user-oriented design. Proceedings of the 19th ICA Conference, Ottawa.

Haeberling, C. (2003). Topografische 3D-Karten:Thesen für kartografische Gestaltungsgrundsätze. Zurich, ETH, Zurich: 255.

Haire, K. (2001). Active Object and Agent Based Approaches to Automated Generalisation. 4th ICA Workshop on Progress in Automated Map Generalization, Beijing.

Hake, G., Grünreich, D., and Meng, L. (2002). Kartographie - Visualisierung raum-zeitlicher Informationen. Berlin.

Hardy, P., Hayles, M., and Patrick, R. (2003). Clarity - a new Environment for Generalisation using AGENTS, JAVA, XML and Topology. ICA Map Generalization, PARIS.

Harrie, L. (2001). Weight setting and Quality Assessment in Simultaneous Graphic Generalisation. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Harvey, F. (1999). Pessimistic Generalization: Limits and Lessons for Semantic Interoperability. ICA Workshop on Map Generalisation, Ottawa.

Haykin, S. (1994). Neural Networks: A Comprehensive Foundation. NY, Macmillan.

Haykin, S. (1999). Neural Networks -- a Comprehensive Foundation,, Prentice Hall,.

Højholt, P. (2000). Integration of Generalization into Map Production: Putting it All Together. ICA, Seminar on On-demand Mapping, Barcelona.

Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. Michigan, University of Michigan.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. Proc. Natl. Acad. Sci. USA,.

Hu, M. K. (1962). "Visual pattern recognition by moment invariants." IRE Tr. on Inf. Theory 8: 179-187.

Jiang, B., and Harrie, L. (2003). Cartographic Selection Using Self-Organizing Maps. ICA Map Generalisation, PARIS.

Jiang, B. (2004). Spatial Clustering for Mining Knowledge in Support of Generalization Processes in GIS. ICA Workshop on Generalisation and Multiple representation; 20-21 August 2004, Leicester.

Jodio, E. M. (1990). "What expert systems don't know: the role of the user in GIS generalization." NATO ASI on "coginitve and linguistic: Aspects pf Geographic space: 1-12.

Jones, C. B., Bundy, G.Li. and Ware, J.M. (1995). "Map Generalization with a Triangulated Data Structure." Cartography and GeographicInformation Systems, 22(4): 317-331.

Kada, M. (2002). Automatic generalisation of 3D building models. Sympsium on eospatial theory, processing and applications, Ottawa.

Keller, S. F. (1995). Potentials and limitations of AI techniques applied to generalization, Taylor & Francis.

Keller, S. F. (1995). Generalization by example: Interactive parameter control in line generalization using GAs. Proc. 17th international cartographic conference, Barcelone.

Keyes, L., and Winstanley, A. (2001). Topographic object recognition through shape. Southampton, UK, Ordnance Survey.

kilpeläinen, T., and Tapani Sarjakoski, T. (1995). Increamntal generalization for multiple representations of geographicval objects, Taylor & Francis.

kilpeläinen, T. (2000). "Knowledge acquisation for generalization rules." Cartography and Geographic information Sciences 27(1): 41-50.

kilpeläinen, T. (2000). "Maintenance of multiple represemtation databases for topographic data." The Cartographic Journal 37(2): 101-107.

Kreveld, M., Oostrum, R.M., and Snoeyink, J. (1997). Efficient settlement selection for interactive display. In Proc. Auto-Carto 13: ACSM/ASPRS Annual Convention Technical Papers.

Kreveld, M. (1999). Twelve Computational Geometry Problems from Cartographic Generalization. ICA Workshop on Map Generalisation, Ottawa.

Lang, F. (1999). Automatic Geometric and Semantic Reconstruction of buildings from Images by Extraction of 3D-Corners and their 3D-Aggregation. Bonn, Institute for Photogrammetry, Bonn University.

Lang, F. (1999). Semiautomatic 3D-Building Reconstruction with a Digital One-Eye Stereo System. Hanover, Institute for Photogrammetry, Hanover University.

Lee, D. (1999). Practical Solutions for Specific Generalization Tasks. ICA workshop on Map Generalisation, Ottawa.

Lee, D. (2001). Moving Towards New Technology for Generalization. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Lee, D. (2003). Recent Generalization Development and road ahead. ICA Map Generalisation, PARIS.

Lehto, L., and Kilpeläinen, T. (2000). "Real time generalization of geodata in the web." Int. archives of photogrammetry and remote sensing 23(B4): 559-567.

Lehto, L., and Kilpeläinen, T. (2001). Generalizing XML-encoded spatial data on the web. Proceeding of 20th International Cartographic Conference, Beijing.

Lemarié, C. (2003). Generalisation process for Top100: research in generalisation brought to fruition. ICA Map Generalisation, IGN, PARIS.

Li, G., Bundy, Chris, B., Jones, and Furse, E. (1995). Hilostic generalization of large scale cartographic data, Taylor and Francis.

Lindeberg, T. (1996). Scale-Space: A framework for handling image structure at multiple scales. In. Proc. CERN, School of Computing.

Lonergan, M. E., and Jones, C.B. (1999). The Deformation and Displacement of Areal Objects During Automated Map Generalisation: An Approach to Conflict Resolution Through Maximising Nearest Neighbour Distances. ICA workshop on Map Generalisation, Ottawa.

Mackaness, W., and Ruas, A. (1997). Strategies for urban map generalisation. S. Conférence de l'Association de Cartographie Internationale, Stockholm.

Malik, J. (2002). "Shape matching and object recognition usoing shape context." IEEE transaction of pattern analysis and machine intelligence 234(2).

Mark, W. J., Thomas, N., and Jones, C. (2003). Resolving Graphic Conflict in Scale Reduced Maps: Refining the Simulated Annealing Technique. ICA, Map Generalisation, IGN, PARIS.

Marr, D., and Nishihara, H.K. (1978). Representation and recognition of the spatial organization of three-dimensional shapes. Proceedings of the Royal Society of London.

Martin, E., Kriegel, H.P., Sander,J., and Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining, Portland.

Mayer, H. (1998). Three Dimensional Generalization of Buildings Based on Scale-Spaces. MÜNCHEN, Chair for Photogrammetry and Remote Sensing, TUM.

McMaster, R., and Robert, B. (1991). Conceptual frameworks for geographical knowledge. In Map Generalization: Making Decisions for Knowledge, Scientific & Technical.

McMaster, R. a. S., S. (1992). Generalization in Digital Cartography. Washington, Associationof American Geographers, Washington.

Mcmaster, R. B. (1986). "A statistical analysis of mathematical measures for linear simplification." The American Cartographer 13(2): 103-116.

McMaster, R. B. (1995). Knowledge acquisation for cartographic generalization, Taylor & Francis.

Meng, L. (1997). "Automatic Generalization of Geographic Data." Tech. Report, SWECO, Stockholm, Svenska Försvarsmakten (Swedish Armed Forces).

Meng, L. (1998). "Strategies on Automatic Generalization of Geographic Data - Cognitive Modeling of Cartographic Generalization." Tech. Report for Stiftelse för Kunskap- och Kompetensutvecklung (Swedish Foundation for Knowledge and Competence development).

Meng, L. (2002). How Should 3D Geovisualization Please User Eyes Better?,. Euroconference on methods to define geovisualizatio contents for users needs, Albufeira, Portugal.

Moin, S., Brault, J.J. (1997). Structure Recognition of On-Line Handwritten Documents Based on SGML,. ICDAR97.

Mokhtarian, F., and Mackworth, A.K. (1992). "A Theory of Multiscale, Curvature-Based Shape Representation for Planar Curves." IEEE Transactions on Pattern Analysis and Machine Intelligence 14(8): 789-805.

Monika, S. (2000). "GENERALIZATION BASED ON LEAST SQUARES ADJUSTMENT." International Archives of Photogrammetry and Remote Sensing 33(4).

Monmonier, M. (1989). "Interpolated Generalization: Cartographic Theory for Expert- Guided Feature Displacement." Cartographica 26(1): 43-64.

Monmonier, M., and Mcmaster, R.B. (1989). Geometric operators and sequential effects in cartographic line generalization. 14th conference of ICA, Budapest.

Monmonier, M. S. (1982). Computer.assisted cartogrphiy.

Mustiere, S., and Moulin, B. (2002). What is spatial context in cartographic generalization. Symposium on Geospatial Theory, Processing and Applications, Ottawa.

Mustiere, S., Mustiere, S., Moulin, B., and Moulin, B. (2002). "What is Spatial Context in Cartographic Generalisation?" IAPRS & SIS, Geospatial Theory, processing and Applications 34(4): 274-278.

Mustière, S., and Duchêne, C. (2001). Comparison of Different Approaches to Combine Road Generalisation Algorithms: GALBE, AGENT and CartoLearn. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

NeuNet (2001). Back propgation training, http://www.cormactech.com/.

Nigrin, A. (1993). Neural Networks for Pattern Recognition. MA, Cambridge.

Paul, S., and Garland, M. (1997). Survey of polygonal surface Algorithms. Pittsburgh, USA, School of Computer Sc., Carnegie Mellon University: 5.

Penev, P. S., and Atick, J. (1996). "Local Feature Analysis: a General Statistical Theory for Object Representation." Network: Computation in Neural Systems 7: 477-500.

Peter, B., and Weibel, R. (1999). Using vector and raster based techniques in categorical map generalization. ICA workshop on progress in automated map generalization, Ottawa.

Peter, B., and Weibel, R. (1999). Using Vector and Raster-Based Techniques in Categorical Map Generalization. ICA workshop on Map Generalisation, Ottawa.

Peter, B., and Weibel, R. (1999). "Using Vector and Raster-Based Techniques in Categorical Map Generalization."

Peter, B. (2001). Measures for the Generalization of Polygonal Maps with Categorical Data. Fourth ICA Workshop on Progress in Automated Map Generalization, Beijing.

Peter, B. (2001). "Measures for the Generalization of Polygonal Maps with Categorical Data."

Pfeiffer, M., and Pandit, M. (1995). A parametrical description of plane curves using wavelet descriptors. proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing.

Pinkus, A. (1999). "Approximation theory of the MLP model in neural networks." Acta Numerica 8: 143-196.

Principe, J. C., Euliano, N.R., and Lefebvre, W.C. (2000). Neural and Adaptive Systems: Fundamentals Through Simulations, John Wiley & Sons, Inc.

Raheja, J., and Meng, L. (2002). "Rules and constraints for 3D generalisation of urban area." Acta Simica Geographica, Beijing 11: 17-28.

Rainsford, D., and Mackaness, W. (2002). Template Matching in Support of Generalisation of Rural Buildings. Symposium on Geospatial Theory, Processing and Applications, Ottawa.

Regnaild, N. (2001). "Contextual building typification in automated map generalization." Algorithmica 30: 312-333.

Regnauld, N., Edwardes, A. and Barrault, M. (1999). Strategies in Building Generalisation: Modelling the Sequence. Third Workshop on Progress in Automated Map Generalization, Ottawa.

Regnauld, N., Edwardes, A. and Barrault, M. (1999). Strategies in Building Generalisation: Modelling the Sequence, Constraining the Choice. ICA workshop on Map Generalisation, Ottawa.

Reichenbacher, T. (2001). "Adative concepts for a mobile cartography." Journal of Geographical sciences 11: 43-53.

Riesenhuber, M., and Poggio, T. (2000). "Nature neuroscience supplement." nature neuroscience supplement 3.

Riesenhuber, M., and Poggio, T. (2000). "Models of object recognition." nature neuroscience supplement 3.

Robinson, A. H., and Petchenik, B.B. (1976). The nature of maps. Chicago, USA, Univ. Chicago Press, Chicago.

Robinson, A. H. (1984). Elements of Cartography, John Wiley & Sons, Inc.,.

Robinson, G. J. (1995). A hierarchical top-down bottem-up approach to topographic map generalization, Taylor and francis.

Robinson, G. J. (1995). A hierarchical top-down bottom-up approach to topographic map generalisation.

Ruas, A., and Plazanet, C. (1996). Strategies for automated generalization. 7th International Symposium on Spatial Data Handling. Delft (Netherlands), Netherlands.

Ruas, A. (1998). Constraint Analysis. Zurich, Dept. of Geography, University of Zurich.

Ruas, A. (1999). The Role of Meso Level for Urban Generalisation. ICA workshop on Map Generalisation, Ottawa.

Ruas, A. (2000). Project AGENT: Overview and Results of a European R&D Project in Map Generalization. ICA, Seminar on On-demand Mapping, Barcelona.

Ruas, A., Plazanet C. (2001). Automating the process of generalisation, The Age of maturity. Conférence de l'Association de Cartographie Internationale, PEKIN.

Ruas, A. (2001). Strategic algorithms for organisations. PARIS, IGN, France: 68.

Sarjakoski, T., and Kilpeläinen, T. (1999). Least Squares Adjustment for Generalization of Large Data Sets. ICA99 Ottawa Workshop on Generalization., Ottawa.

Sarjakoski, T., and Letho, L. (2001). Challenges of Information Society for Map Generalization. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Schyns, P. G., Goldstone, R.L., and Thibout, J. (1998). "The Developemt of fetures in object concepts", Behavioral and Brain." Sciences.

Schyns, P. G. (1998). "The development of features in object concepts." Behavioral and Brain sciences 21: 1-54.

Sébastien, M., Zucker, J.D., Saitta, L. (1999). Cartographic Generalization as a Combination of Representing and Abstracting Knowledge. Proceedings of the seventh ACM international symposium on Advances in geographic information systems, Kansas City, Missouri, United States.

Seng, T. S. (2000). Neural Networks. Singapore, National University of Singpore.

Serra, J. (1982). Image Analysis and Mathematical Morphology. London, Great Britain., Academic Press.

Sester, M. (2001). Kohonen Feature Nets for Typification. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Sester, M. (2001). Kohonen Feature Nets for Typification. Fourth Workshop on Progress in Automated Map Generalization., Beijing.

Sester, M. (2002). "Application Dependent Generalization - The Case of Pedestrian Navigation." Geospatial Theory, Processing and Applications" 34, Part 4.

Sester, M. (2003). 3D-Generalization and Abstraction, Beitrag zum Seminar Computational Cartography and Spatial Modelling. International Conference and Research Center for Computer Science, Schloss Dagstuhl, Wadern, Germany.

Skopeliti, A., and Tsoulos, L. (2001). A Methodology for the Assessment of Generalization Quality. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Skopeliti, A., and Tsoulos, L. (2001). "A Methodology for the Assessment of Generalization Quality."

Steven, V. D., and Thierens, D. (1998). Robust genetic algorithms for high quality map labeling. Utrecht, Utrecht University.

Su, B., Li, Z., Lodwick, G., and Müller, J.C (1997). "Algebraic models for the aggregation of area features based upon morphological operators." International Journal of Geographical Information Science 11(3): 233-246.

Suveg, I., Vosselman, G. (2001). "3D building reconstruction by map based generation and evaluation of hypotheses."

Thiemann, F. (2002). "Generalization of 3D Building Data." IAPRS, Geospatial Theory, Processing and Applications 34(4).

Thiemann, F., and Sester, M. (2004). Segmentation of buildings for 3D-generalisation. ICA Workshop on Generalisation and Multiple Representation,, Leicester, UK.

Thomson, R., and Brooks, R. (2000). Efficient Generalisation and Abstraction of Network Data Using Perceptual Grouping. Fifth International Conference on GeoComputation, UK.

Tieng, Q. M. (1997). "Boles Recognition of 2D Object Contours Using the Wavelet Transform Zero-Crossing Representation." IEEE Trans. on PAMI 19(8).

Töpfer, F., and Pilliwizer, W. (1966). "The Principles of Selection." The Cartographic Journal 3(1): 10-16.

Van, P., and Jones, C. (2001). Feature Based Line Generalisation Using Delaunay Triangulation. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Vasilis, D., and Hedzilacos, T. (1997). On the Assessment of Generalization Consistency, advances in Spatial Databases. 5th International Symposium, SSD'97, Berlin, Germany.

Visvalingam, M. (1999). Aspects of (Line) Generalisation. ICA workshop on Map Generalisation, Ottawa.

Vranic, D. V., and Saupe, D. Shape Descriptor Based on 3D Fourier Transform. Leipzig,, Germany, University of Leipzig,, Germany.

Wallace, T. P., and Wintz, P. A (1980). "An Efficient Three-Dimensional Aircraft Recognition Algorithm Using Normalised Fourier Descriptors." Computer Graphics and Image Processing 13: 99-106.

Wang, J. Y., and Cohen, F.S. (1994). "3-D object recognition and shape estimation from image contours using B-splines, shape invariant matching, and neural network." IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on 16(1): 13-23.

Ware, M., Jones, C., and Thomas, N. (2001). A Simulated Annealing Algorithm for Cartographic Map Generalization with Multiple Operators. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Weibel, R. (1991). Amplified Intelligence and Rule-based Systems, Map Generalization: 172-186.

Weibel, R. (1991). Specifications for a platform to support research in map generalization. 15th conference of the ICA.

Weibel, R., and Buttenfield, B. (1992). "Improvement of GIS graphics for analysis and decision-making." Int. Journal of GIS 6(3): 223-245.

Weibel, R. (1995). An evaluation of MGE Map Generaliser. Zurich, University of Zurich.

Weibel, R., Keller S., and Reichenbacher, T. (1995). Overcoming the Knowledge Acquisition Bottleneck in Map Generalization :the Role of Interactive Systems and Computionnal Intelligence. Proceedings of COSIT.

Weibel, R., and Müller, J.C. (1995). Generalization: State of the art and issues, Taylor & Francis.

Weibel, R. (1995). Three essential building blocks for automated generalization, Taylor and francis.

Weibel, R. (1997a). Generalization of Spatial Data: Principles and Selected Algorithms, Springer, Berlin.

Weibel, R., and Dutton, G. (1998). Constraint-Based Automated Map Generalization. 8th International Symposium on Spatial Data Handling.

Weibel, R. (2000). Challenges for Generalization and Web Mapping. ICA, Seminar on On-demand Mapping, Barcelona.

Werschlein, T., and Weibel, R. (1994). USE OF NEURAL NETWORKS IN LINE GENERALIZATION, EGIS.

Winstanley, A. C. (1998). Structuring Vector Maps using Computer Vision Techniques. Conference of the Association of Geographic Information, Birmingham.

Wunch, P., and Laine, A. (1995). "Wavelet descriptors for multi-resolution recognition of handprinted charrecters." Pattern Recognition 28(8): 1237-1249.

Yang, H. S., Lee, S., and Lee, K.M. (1998). "Recognition of 2D Object Contours Using Starting-Point-Independent Wavelet Coefficient Matching." Journal of Visual Communication and Image Representation 9(2): 171-181.

Zabrodsky, H. (1992). A Measure of Symmetry Based on Shape Similarity. Proc. IEEEE-CVPR-92, Champion.

Zhou, S., and Jones, C. (2001). Multi-Scale Spatial Database and Map Generalisation. 4th ICA Workshop on Progress in Automated Map Generalizatio, Beijing.

Zurada, J. M. (1992). Introduction To Artificial Neural Systems. Boston:, PWS Publishing Company.

Appendix: Abbreviations

- SR: Structure Recognition
- ANN: Artificial Neural Network
- 2D : Two Dimensional
- 3D : Three Dimensional
- Geons : Geometrical ions
- GSD : Geo Structural Description
- LSA: Least Square Adjustment
- LoD: Level Of Detail
- Brep: Boundary Representation
- CSM: Continuous Symmetry Measure
- NN: Nearest Neighborhood
- RMS: Root Mean Squares
- MST: Minimum Spanning Tree