SKETCHING SPATIAL QUERIES

By

Andreas D. Blaser

Dipl.-Ing. ETH, Swiss Federal Institute of Technology in Zürich, 1992

A THESIS

Submitted in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy
(in Spatial Information Science and Engineering)

The Graduate School
University of Maine

May, 2000

Advisory Committee:

Max J. Egenhofer, Professor in Spatial Information Science and Engineering, Advisor

Peggy Agouris, Assistant Professor in Spatial Information Science and Engineering

M. Kate Beard-Tisdale, Associate Professor in Spatial Information Science and Engineering

Douglas M. Flewelling, Assistant Professor in Geography, State University of New York, Buffalo

Werner Kuhn, Professor in Geoinformatics, University of Münster, Germany

Acknowledgments

Although this thesis bears only one author's name on its cover it could–like most any other complex work–not have been achieved without the hefty support of a great number of people. I am very thankful for this help and would like to thank you all. Another reason that made writing this thesis such a great experience was certainly the spatial location of our University. People here in Maine are friendly, and the environment and the pace of life are very healthy, considered from many perspectives and especially for families. This was certainly also a reason why my family and I had (have) such a good time during our stay in the New World. In this context and above all, I would like to thank Nadia, Erik, and Sven for daring the great crossing with me, for fighting the odds of a new environment, supporting me in many ways, but also for enjoying this unique experience with me.

On the academic side my special thanks go to Max Egenhofer. He was not only head of my advisory committee but also the source of ideas for this thesis as well as my mentor in many ways. I thank him for guiding me and for leaving me a lot of free room so that I could try out new things and experiment. Nonetheless, I would like to thank him for his sparking ideas, inspiring discussions, his trust and belief, and his positive support throughout the entire process of the thesis. A very special thanks goes also to Werner Kuhn, who directed my attention to UMaine and who was the actual reason that I started the PhD in the first place. It was also after an hour-long phone conversation with Werner that I thought I might want to try a next step in academia. Many thanks go also to the other members of my committee, to Kate Beard, Peggy Agouris, and Doug Flewelling. They were always available for my questions and their suggestions and comments helped me a lot to stay on my path and gave me confidence that I was doing the right thing.

The SIE Department and all the people that work, research, and live there deserve another special word of thanks. This exciting, multi-cultural composition of people together with just the right size of the department, provide a perfect foundation for writing a thesis. In this context I would like to thank you all. Special thanks go to mui dos amigos Brasilieros Miro and Fred for being such great office mates and friends, to Yves my Franco-Swiss office mate, neighbor, and buddy for his enthusiasm and motivation during his stay in Orono, and to Martin Raubal, my other office mate, who left as Caporopa for Austria to terminate some unfinished business.

I would also like to thank Andrea for the many discussions about anything from teacups and family life to the ontology of ontology, Doug from block N4 for the dialogues ranging from outer space to the two national Bills that frequently ended with a comparison of apples and fences. Thanks also to Troy for having always an open ear for my special requests, Sotiris for his inspiring Greek cheerfulness, Thomas and Craig for the many early bird chats, Kathleen, Roop, Isolde, Michela, James, David, Valerie, Pete, Paul, Chris, Stephanie, Ashton, Alfred, Harlan, Toni, Karen, and Blane as well as to all the other folks of the department for making Boardman Hall such a nice and familiar place to be.

Outside the department I would like to thank my brother, Oliver for being such a reliable link and beacon back in Switzerland, my parents, Ursel and Hans Peter for bringing me up in a caring environment and learning me to be curious about the world, and Ariane for the many sparking e-mail conversations. My thanks go also to Bidli, Chrigu, Bettina, Res, Tom, Kathrin, Tilli, Brigitte, Lisa, Reto, Marc, Peter, Moni, Rashid, Werner, and Marianne for staying in touch with us while we were abroad. Thanks also to Don, Mildred, Lori, and Dan for sharing their enthusiasm for flying with me, Jayant and Maduh for being such good neighbors at UPark and introducing us to the Maine-Basics, to Jonathan Gregory, for his goodwill also when I was a little bit late, and last but not least to Steffi and Marici for their ever caring friendship and support.

Finally, my thanks go to the Air Force Research Laboratory and the National Science Foundation who funded this research in part under grant no. F 30 602-95-I-0042 and IRI-9613646, respectively.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Sketching is an old form of communication that has been used to visualize, record, and exchange information for thousands of years. Despite its proven expressiveness, it has not yet become a frequently used modality to interact with computer systems. Geographic information systems (GISs) have a particular need for such advanced forms of user interaction, because they can involve complex and heterogeneous data structures that are frequently difficult to describe.

The objective of this thesis is to investigate the usefulness of sketching to retrieve spatial information and to develop methods and an efficient model to capture a sketched spatial scene so that it can be used to query a spatial database. In this scope, we explore the sketching behavior of people and develop a compact object-oriented representation for freehand sketches. A prototype application of a sketch-based query system is implemented to verify the developed concepts and theories.

1.1 Spatial Information Retrieval

Information is the meaning of data after it has been interpreted by people. Because data consist primarily of recorded facts, it is important to have knowledge about the context of data to comprehend its meaning. Spatial information adds a spatial reference or component to this notion. During the last 20 years, spatial information has become increasingly important and people have realized that it is frequently more appropriate to reference information spatially than in abstract forms. This tendency promoted the development of systems that allow people to capture, check, integrate, manipulate, analyze, and display spatial information (Goodchild 1987; Laurini and Thompson 1992). Systems with such capabilities are referred to as *geographic information systems*.

Early GISs were primarily used to produce maps, demographics, or other geo-indexed charts. Current GISs are much more versatile and have found their way into application areas previously unknown to GIS (Coppock and Rhind 1991; Maguire *et al.* 1991; Longley *et al.* 1999). This diversification also changed the composition of the GIS user community. People using GISs belonged to a rather small elite of technically trained engineers or technicians; however, today a typical GIS has to accommodate a large spectrum of user skill levels, ranging from casual users to GIS professionals. It is, therefore, crucial that a GIS provides appropriate means of interaction that allow all users access to spatial information.

Storing, managing, and analyzing information are important issues for every GIS; however, information is only useful if it can be retrieved. The process of information retrieval consists essentially of three steps. The first step is the *query formulation*. At this point the user defines what he or she is looking for. The next step requires the system to browse through available datasets and *search* for configurations that match with the user's request. If matches were found, then the *result presentation* completes the retrieval process.

It is possible to define specific system requirements, for each of these steps. The query formulation, for instance, must be simple, intuitive, yet expressive so that a precise question can be asked. During the search phase, the focus is on accuracy and efficiency, that is, the result of the search must match with the query and the user's expectations. The efficiency of a search depends on how the content of the query is compared with information stored in the system. The result presentation is the final step in retrieving information. At this point it is essential to present the retrieved results in an understandable form, without distracting a user with irrelevant details.

1.1.1 Interaction

Interaction is the process of two or more entities influencing each other. In computer science the term *interaction* is frequently used to describe the communication between user and computer and referred to as human-computer interaction (HCI) (Helander 1988). HCI is a relatively new research direction and up to the present, there is no consent about the definition of the range of topics that form the field of HCI (Hewett *et al.* 1992). This lack of agreement results partially from the various perspectives of HCI that scientists with different backgrounds have. This group of people includes computer scientists and psychologists, as well as researchers from cognitive science, such as designers, engineers, or architects. The working definition for HCI formulated by the Special Interest Group on Computer-Human Interaction of the Association for Computer Machinery comes from the perspective of computer science (SIGCHI 1992):

> *Human-computer interaction is a discipline concerned with the design, evaluation, and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them.*

Under these considerations, both *query formulation* and *result presentation* can be associated with HCI (Blaser *et al.* 2000). However, a discussion of both topics is out of scope for this thesis and, therefore, we focus on issues of query formulation alone.

1.1.1.1 Early Forms of HCI

HCI emerged with the use of the first cathode-ray tube (CRT) monitors providing users with a tool to visually interact with computers. In the early 1960s, after exploiting possibilities of solely text-oriented applications, a new discipline of computer science was born: *Computer Graphics*. Many techniques of that field originate in Sutherland's *Sketchpad* (Sutherland 1963), which was an early computer aided design (CAD) application that allowed a user to construct, edit, and replicate geometric patterns on a computer screen. Ideas of automatic error correction, such as snapping start and end points of a line, were introduced, as was the use of a light pen for input generation.

Another milestone in the development of human-computers interaction was the *Dynabook* (Kay and Goldberg 1977), a portable interactive personal computer with wireless communication and a flat panel display that was as accessible as a book. Other important inventions, such as the computer mouse, bitmapped displays, the technique of windowing, the desktop metaphor, and the principle of object-oriented programming (Goldberg and Kay 1976; Goldberg and Robson 1983), can also be traced back to these ideas.

Improvements in HCI have led to an enhanced usability and a broad acceptance of computers in our everyday life. Today we can find computing devices on almost every desk and in almost every household, which can be attributed to fundamental innovations in human-computer interaction technology. Many relevant publications in this context can be found in the Handbook of Human-Computer Interaction (Helander 1988).

1.1.1.2 HCI in the Late 1990s

User interfaces have made remarkable improvements over the last 20 years (Myers 1996). Graphical user interfaces of modern operating systems are out of the box object-oriented, which means that most perceptible objects can be directly manipulated (Shneiderman 1983; Hutchins *et al.* 1986). These user interfaces have a consistent look, are mainly visually oriented, and they can host a multitude of different applications under the same hood. Today's user interaction involves primarily typing with a keyboard and selecting or drawing with a pointing device, such as mouse, trackball, or touchpad. The use of alternative input devices, such as pens, is limited to special application, such as palm computing, CAD, or professional design (Greenstein and Arnaut 1998).

1.1.1.3 Multi-Modal User Interaction: The Future of HCI

The future development of HCI is expected to be characterized by new, innovative, and human-centered input devices that are made possible through the advent of more powerful computers and improved interaction techniques (Shneiderman 1990; Vo *et al.* 1995). These user interfaces will be guided by the principles of direct manipulation (Shneiderman 1997), individual customizability, and they will be based on intuitive metaphors (Wilson 1990). They will incorporate techniques, such as voice or gesture recognition (Wexelblat 1995), and the use of pens will be commonplace (Mel *et al.* 1988). Some user interfaces will explore three-dimensional space (Leach *et al.* 1997). This new era of HCI will greatly simplify the way people interact with computers and it will pave the way for intelligent and portable devices that will change our relationship to computers completely (Egenhofer and Kuhn 1998). Computing devices will continue to get smaller and they will provide better performance and enhanced connectivity at lower cost. This, in turn, will lead to an assimilation of computers into almost every human domain, involving people previously left out of the computer revolution (Hewett *et al.* 1992).

The methods that people use to interact with computers are referred to as user modalities. Modalities address any type of sensation, including vision, hearing, and various ways of expression, such as writing, talking, or gesturing. A multi-modal interaction involves more than one form of communication at the same time (Kuhn 1992; Waibel *et al.* 1995). The use of multiple modalities increases the flexibility and the reliability of a user interface, because people can choose the method of interaction according to their preferences, skills, and the task to accomplish (Oviatt 1999). In systems that deal with spatial information, multi-modal interaction can lead to an increased efficiency as well (Oviatt 1997). Another benefit of multi-modal systems is that they support a greater flux of information between user and computer (Blaser *et al.* 2000). Future computer devices are likely to incorporate multi-modal user interfaces, because this form of interaction comes closer to the way people interact with each other.

Many future computing devices will be used primarily to access, retrieve, and analyze information. Access to information must be simple, fast, and efficient (Lewis 1995), which requires specially adapted user interfaces and appropriate means of interaction (Egenhofer and Kuhn 1998). Research in HCI is, therefore, crucial in this context and resulting innovations will have a deep impact on our society.

1.1.1.4 Sketching—An Alternative Modality for GIS

User interaction in GIS today is not much different from that in other application domains (Draper 1996). A user's primary tools are keyboard and mouse, and occasionally a digitizing tablet. Many common tasks are executed via pull-down menus or predefined buttons in dialog boxes. Unfortunately, this kind of interaction between person and machine is often unintuitive and cumbersome, which results in high training costs and many operational errors. Relying on traditional modalities to improve this situation appears inadequate, because pointing and typing are not expressive and flexible enough (Egenhofer 1990; Egenhofer 1996a).

The integration of alternative modalities into GIS user interfaces appears to be a promising approach to improve this situation, notably for applications where spatial information must be retrieved (Egenhofer and Kuhn 1999). Two particularly interesting user modalities are *sketching* and *talking*. People use and practice these modalities daily, such that they develop a high level of expertise. Both modalities have specific advantages for certain application areas; however, sketching is especially well suited to describe spatial scenes (Blades 1990; Oviatt 1997). This characteristic is of particular interest for the retrieval of spatial information, since most current query methods rely on the formulation of textual query languages.

Textual query languages *per se* are non-spatial. As a result, it is often difficult and unintuitive to formulate a spatial query textually (Egenhofer 1992). Sketching, on the other hand, is a direct, creative, and visual form of expression (Goldschmidt 1991). Objects can be represented in two-dimensional space and characteristics, such as shape or orientation, can easily be rendered. Non-spatial attributes and properties can be assigned through either written or spoken annotations. The base for a logical structure is implicitly given by the arrangement and the composition of objects within the sketch. Simply placing two objects close to each other, for instance, can indicate a neighborhood relationship between the objects. Relations, in this context, can be spatial, hierarchical, or conceptual and sketched objects can be physical or virtual (Blaser *et al.* 2000).

A sketch reflects a spatial scene in a more objective way than any verbal description, because the spatial information about the scene is directly accessible. As a result, sketches are less susceptible to spatial interpretation errors and ambiguities. Because of their visual and clear nature, sketches are also well suited to describe *complex* spatial situations. Objects can be annotated, aggregated to larger entities, or graphically emphasized. The use of symbolic or diagrammatic representations can further enhance the semantics of drawn objects. Therefore, people familiar with only a small set of sketching concepts can easily understand also complicated sketched situations.

Sketching and talking are complementary to traditional forms of user interaction. An ideal user interface for GIS must, therefore, combine multiple modalities, allowing a user to choose the method of interaction according to the user's knowledge, skill, and liking, and depending on the particular situation (Wilson 1990).

1.1.2 Processing

The term *processing* stands for a series of actions, manipulations, or functions generating results. At the heart of every *information retrieval process* are basic operations that *compare* the content of the query statement (i.e., question) with data in a database. The methodology used for this comparison depends on the query formulation, the kind of data available, and the intended purpose of the query. There are approaches that specialize on comparing local or global characteristics of datasets and others that focus on attributes or properties of distinct entities. The former approaches can be described as *field-oriented* and the latter as *content-oriented*. Mixed forms are possible as well. The distinction between field-oriented and content-oriented approaches is often fluent, because the term *content* is frequently interpreted differently. We refer to *content* as the meaningful part of a distinct entity that has semantics of its own and can be interpreted as an object.

Retrieving information may involve additional processes besides comparing. Examples are preprocessing and interpreting the query formulation, or ranking and post processing retrieved results. The efficiency of the information retrieval process depends on the complexity of the query, the size of the database, and the efficiency of individual sub-processes.

1.1.2.1 Matching Characteristics

Databases that contain unstructured or uninterpreted data are frequently queried for specific data characteristics. These databases may contain aerial photographs, satellite imagery, or other field-type data, such as digital terrain models or hydrological flow models. The query formulation is based on field characteristics, such as color, hue, texture, or a specific pattern. A query may also involve the size or location of clusters in a field. Figure 1.1 shows two examples of field-based queries.

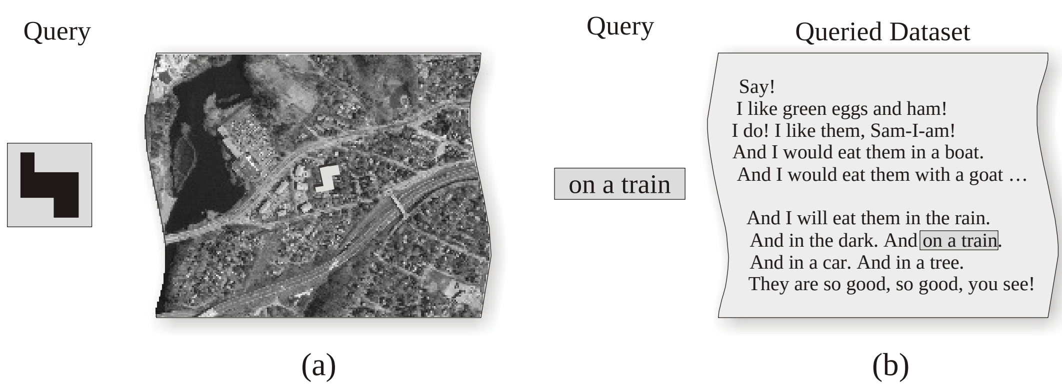


Figure 1.1 Two schematic field-based queries: (a) an aerial photograph and (b) a scanned text page, extracted from Geisel and Geisel (1960).

The objective of using such an approach is to match a query statement with data in a database. To find a valid match, an application has to scan through the entire field of pixels and search for distributions similar to that in the template. Depending on the implementation a template can be relaxed, that is, certain characteristics of the template, such as size or orientation, can be changed during the matching process. This approach works well if template and data are very similar. However, if one wing of the searched building on the photograph has the same color as the background then the query will not produce a satisfying result.

1.1.2.2 Matching Entities

Databases that contain structured data with an associated meaning and distinguishable entities can be queried for their content (content-based query). Information of this type can be stored in relational databases, tables, or structured computer files. The dataset displayed in Figure 1.2a is a list that reflects the interpreted content of the aerial photograph in Figure 1.1a.

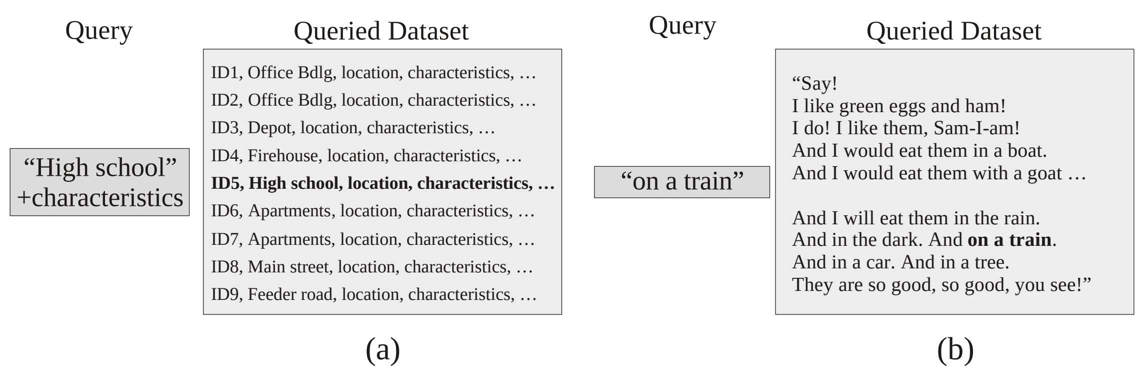


Figure 1.2 Two schematic queries searching for a specific object in the database.

The associated query consists of an object's type and an additional object characterization. In Figure 1.2b the database is a simple text file and the query is a combination of three words. Both cases require the system to search for matching entities and to compare specified characteristics, if such are provided. Depending on the implementation, variations of the query can be retrieved as well. Matching entities or their characteristics is typically more efficient than matching fields (Section 1.1.2.1), because a comparison can be made on a higher level. However, depending on the nature of the queried data, some form of pre-processing may be involved. Queries that focus on entities are also less prone to data misinterpretations, because the concept and the characteristics of an entity instead of the characteristics of a field are queried. For the same reason content-based queries are also easier to relax. For this purpose, an application can reduce the weight of individual query components or it can substitute components with semantically similar components (Rodríguez *et al.* 1999).

1.1.2.3 Matching Groups of Entities

The third form of a query involves two or more distinct entities, which enables an application to take into account relations between entities. Depending on the implemented approach, such relations can be of spatial, temporal, or conceptual nature. The two queries in Figure 1.3 carry such constraints in addition to content information.

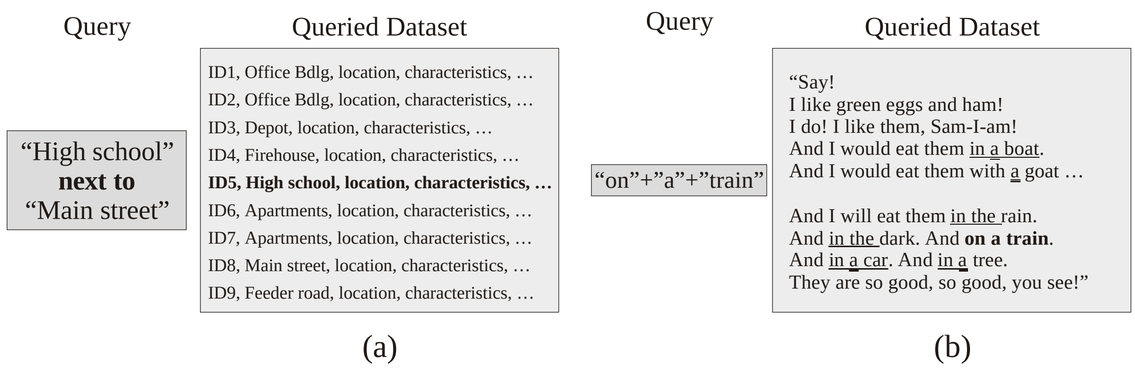


Figure 1.3 Two schematic content and relation-based queries.

The query statement in Figure 1.3a includes a spatial relation and that in Figure 1.3b two conceptual relations. Like with the previous example (Figure 1.2) the system has to match entities; however, in this case the constraints between objects (relations) have to be matched. By relaxing individual constraints, configurations similar to the query can be found. For instance, the three successive words “in a car” in Figure 1.3b are conceptually similar to the query “on a train” (e.g., if the car and the train are both considered as means of transportation).

Queries that are based on matching entities *and* their interrelations offer the same flexibility as queries based on matching entities alone. Since the relationship between entities is taken into account, it is possible to retrieve complex configurations more accurately. A spatial query that does not consider the relationships between entities provides only evidence that queried entities are present, but it does not reveal their spatial arrangement.

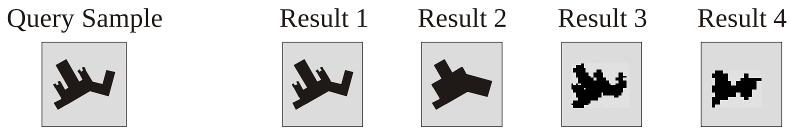
1.1.2.4 Combined Methods

Combined methods are frequently used if unstructured data must be analyzed at a higher level. For this purpose the original dataset has to be preprocessed using field-based methods. The aerial photograph in Figure 1.1a, for instance, could be searched for particular shapes that are extracted and manually interpreted, or classified according to a set of predefined templates. The resulting metadata can be stored conveniently in a list, similar to that in Figure 1.2a. In a subsequent phase the extracted entities (e.g., buildings and streets) can be matched, using an approach similar to that in Figure 1.2a or 1.3a.

1.1.2.5 Exact and Partial Match

The result of a comparison depends on the chosen method. There are those methods that lead to a binary result and those that allow a more distinctive differentiation. The former methods are primarily focused on existence, while the latter provide a measure to estimate the likelihood that the retrieved result matches with the query statement.

Methods that produce a binary result can be used when there are only two results possible (e.g., a pixel that is either black or white) or when a decision-making process requires an unambiguous outcome. A calibration, based on predetermined thresholds, becomes necessary if the result of the matching process is ambiguous. Method A and Method B in Figure 1.4 produce a binary result, but use different thresholds; consequently the similarity values for the second and third result are different. Method C computes a gradual similarity value that can range from 0 (no similarity) to 1 (perfect similarity).



Similarity Values:

	Result 1	Result 2	Result 3	Result 4
Method A	1	1	1	0
Method B	1	1	0	0
Method C	0.99	0.75	0.50	0.03

Figure 1.4 Comparison of three field-based matching methods.

Matching methods that evaluate a gradual correspondence between entities or characteristics are more flexible, because they reflect the actual similarity more accurately. Since many processes require an unambiguous outcome, gradual matching methods are frequently used to assess the similarity between individual components within the scope of a more general comparison. These individual similarity values are then combined and the result can be compared against a threshold, producing a binary result.

1.1.2.6 Efficiency

Efficiency is the quality of a system to perform a task quickly, that is, the ratio of the effective or useful output to the total input in any system. The efficiency of two different systems, sharing the same initial settings and generating the same or similar results, can be evaluated by comparing the processing times for these tasks. This comparison allows a *relative assessment of efficiency* between systems. This approach is often more appropriate than evaluating an absolute efficiency for a system, because an isolated efficiency value can be hard to relate within a specific context.

The efficiency of a retrieval process depends on such factors as the chosen comparison method, the type of data compared, the grade of optimization of algorithms, the complexity of the query statement, and the nature of the database. Another factor that influences the efficiency of a retrieval system significantly is concerned with the necessary number of comparisons.

Every individual comparison requires a certain amount of time. Hence, a system that processes a query using 100 comparisons with a specific complexity is always more efficient than a system that has to execute 1000 comparing operations of the same complexity. The fundamental idea is, therefore, to minimize the number of comparisons while maintaining the quality of the result. Often such an optimization is possible. The key lies in an appropriate use of heuristics and *a priori* knowledge. People, for instance, perform well in comparing complex structures, because they rely on highly selective approaches that allow them to focus on relevant details. Irrelevant or impossible comparisons ore omitted. Computers, unlike people, are frequently programmed to compare every possible combination. To increase the efficiency of a retrieval system it is, therefore, important to implement methods and algorithms that execute only a relevant subset of all possible comparison operations for a specific query.

1.2 A Sketch-Based System for Querying Spatial Information

Sketching promises to be an appropriate modality to describe spatial information (Section 1.1.1.4), for instance, to formulate a spatial query. This thesis attempts to provide evidence for this assertion. The following sections describe goal, hypothesis, and four central research questions of this research.

1.2.1 Goal

The goal of this thesis is to create the theoretical foundation for a sketch-based system to query spatial information. In this context we want to demonstrate the suitability of sketching as an intuitive and expressive modality to create spatial queries. Another goal of this thesis is to prove the practicability of Spatial-Query-by-Sketch (Egenhofer 1996b) and to integrate spatial formalisms resulting from previous research activities into one system.

1.2.2 Concept

The process of interaction within a sketch-based query system can be described at five conceptual stages. The first phase is concerned with *interpreting* the sketched representation of a spatial scene so that individual, semantic entities of a sketch can be distinguished. Once these entities have been extracted, the system has to *translate* the sketch into a processable form. This process is based on the set of extracted entities, their properties, and a set of formalization rules. The result of this translation is a symbolic representation of the sketched query, called the *Digital Sketch*.

The subsequent processing step involves the *comparison* of the digital sketch with corresponding data in a database. Comparing only relevant characteristics, while omitting unlikely configurations, increases the efficiency of this step. If acceptable matches are found, the system has to *prioritize the results*, such that the most likely result can be presented to the user first. The *result presentation* concludes the process of querying spatial information with a sketch.

1.2.3 Hypothesis

This research focuses on spatial databases that store information as objects and that maintain specific spatial relations among these objects (Elmasri and Navathe 2000). The number of binary relations necessary to describe a spatial situation becomes relevant when a system has to process large spatial configurations with many objects. Since the number of possible binary relations increases exponentially in function of the number of involved entities by $O(n^2)$, it is undesirable to consider all possible combinations of binary spatial relations. Hence, finding a method that reduces the number of relations without decreasing the model's expressiveness is an important foundation for a sketch-based query processor.

The hypothesis of this thesis is, therefore, concerned with finding a specific subset from all spatial relations that, when used to query a sketch database, leads to a similarity assessment comparable to that obtained with the complete set of spatial relations. The focus is on the first few ranks of the retrieved sketches, because they capture the most similar sketches compared to the sketched query. This leads to the formulation of the following hypothesis:

> *There exists a reduced set of spatial relations, much smaller than n^2, that produces an ordering of the significant portion of the ranking list similar to that of the complete set of relations.*

1.2.4 Research Questions

Question 1: *Is a multi-modal GIS user interface feasible?*

We are interested in comparing conventional with alternative user modalities. What are the tasks that a user has to perform in a typical GIS environment? Are there modalities that are especially suitable for certain tasks? These questions are relevant, because their answers may indicate that the current interaction methods are sufficient, that they are inadequate, or that the applicability of modalities depends on the application's context.

Question 2: *How do people sketch?*

Unlike writing and talking, sketching is not a well-defined language with well-understood grammatical rules and a set of predefined structures. The unconstrained form of expression of sketches will influence the methods that are used to extract meaning from sketches. Other important issues concern people's sketching habits and strategies, such as their sketching styles or the sequences with which they draw objects. Observations in this context are crucial for any automated process that attempts to extract spatial information from sketches.

Question 3: *How to efficiently translate a spatial sketch into a digital sketch?*

To automatically process a spatial sketch it is essential to capture the significant portion of a sketched scene in a well-defined manner. This is done by translating the sketched scene into a digital sketch, consisting of elementary building blocks and a description of their interrelations. This translation process is based on methods and algorithms that are designed to capture characteristics of sketched input. The efficiency of these methods becomes increasingly important when large datasets are processed. Their careful design is, therefore, crucial for an effective sketch-based spatial information retrieval system.

Question 4: *Are the proposed formalisms and concepts of a sketch-based query system feasible?*

This question is concerned with the practicability of concepts and formalisms that are developed within the scope of this thesis. A software prototype becomes the test bed for sketch-based interaction, empirical tests of the proposed spatial formalisms, and the overall concept of Spatial-Query-by-Sketch.

1.3 Approach

This thesis is concerned with creating the foundation for a sketch-based system to query spatial information (Section 1.2.1). For such a system to materialize there is research required in different areas: First evidence has to be found that *sketching is a promising alternative* user modality to interact with a spatial information system. A study and evaluation of visual information retrieval systems will reveal how these systems retrieve (spatial) information and where improvements can be made. This analysis, together with an evaluation of different traditional and alternative user modalities, will focus on elaborating those aspects that are key to a successful interaction.

The results of this analysis provide the foundation for an investigation that will focus on the *ontology of sketches*. The basic elements of a sketch and the sketch creation process have to be investigated. Because sketching is an individual form of expression, involving primarily human subjects, we will study *the sketching habits of people* through a survey. This survey will capture how people sketch, what strategies they use, where these strategies converge and where they are different, and most importantly how spatial information is portrayed in geo-spatial sketches. To extract useful information, it is appropriate to statistically analyze the survey. Because no automatic analysis tools are available, this evaluation will be conducted manually, on an object-by-object basis for each surveyed sketch.

The results of this survey provide the foundation for the next phase of this research, the *translation of a freehand sketch into a digital format* that can be computationally processed. This digital footprint of a sketch–*the digital sketch*–contains all relevant aspects of its original and to provide a suitable framework to interconnect its components (*sketched objects*). While we rely on geometric characteristics to portray sketched objects, we will introduce a set of spatial relations to describe binary spatial relations between sketched objects. These spatial relations consider the topological, metrical, and direction relations between individual object pairs (Egenhofer 1996b). To improve the efficiency of the digital sketch we will investigate methods for selecting a relevant subset from all binary spatial relations between sketched objects in a sketch. The result of this investigation is a reduced framework for the digital sketch, called the *reduced association graph*, which interconnects all sketched objects in a specific way.

The next phase is the *translation of the theoretic foundation into a working prototype*. For this purpose we will implement a sketch-based system that allows a user to draw and process a sketched query. The digital sketch can be compared with digital sketches extracted from other spatial scenes (e.g., sketches) and their similarity–*the scene similarity*–can be assessed. This similarity assessment is based on a comparison of individual model components of the digital sketch (e.g., topology, metric, direction, or geometry) (Bruns and Egenhofer 1996). If a sketched query is compared with multiple spatial scenes, then a ranking list can be generated, indicating the similarity of individual spatial scenes with the sketched query. The prototype implementation provides, therefore, an experimental platform for the evaluation of different configurations of the digital sketch.

Comparing the ranking lists, resulting from the similarity assessments based on the reduced association graph and the complete association graph, we obtain evidence if the two approaches produce correlating results (i.e., similar ranking lists). This evaluation considers several datasets to produce a reliable result. The assessment of the correlation between two ranking lists is based on a statistical evaluation.

1.4 Major Results

Binary spatial relations play an important role in spatial object configurations, such as a sketch or a spatial object database. Since the number of binary spatial relations increases by $O(n^2)$ for objects added, it is important to discriminate relevant binary spatial relations from those that are less essential. We have found a method to unambiguously create a relevant subset of all binary spatial relations that has a linear growth. The approach used considers only those binary relations that are established through spatial neighborhood between sketched objects. Through empirical tests we have shown that this reduced set of binary spatial relations provides an appropriate framework for a spatial scene to compare individual spatial scenes according to their similarity. The combination of compactness and relevance of this model to represent the spatial configuration of a spatial scene is significant, because it provides systems that manage large amounts of spatial object-oriented data with a method to store and process spatial relations efficiently.

The implementation that was used for the empirical evaluation of the different approaches to represent, store, and process sketches is based on the concept of Spatial-Query-by-Sketch. The implementation of this concept demonstrated that (1) querying spatial information with sketches is a viable alternative to traditional query methods, (2) the incorporated spatial formalisms are suitable to compare spatial scenes according to their similarity, and (3) the concept of Spatial-Query-by-Sketch is a practicable approach for a sketch-based system to query spatial information.

The survey about the sketching behavior of people provided fundamental knowledge about the ontology of geo-spatial sketches. It showed that typical sketches consist of a small number of simple and abstract geometric figures. People prefer objects, such as closed boxes and straight lines that are arranged in a map-like manner without taking topographic features into account. Written annotations are frequently used to describe, augment, or clarify the semantics of sketched objects. The survey confirmed also previous research, suggesting that people rely primarily on topology to portray spatial scenes; metric and direction relations between objects were used at a secondary level for refinements (Egenhofer and Mark 1995). Besides the predominant topological concept of *disjoint* relations, people were frequently found using topological *meet* and *overlap* relations. Other frequently used spatial concepts among neighboring objects include *parallelity*, *orthogonality*, and an *alignment* of objects with a particular and predominant direction (e.g., one of the cardinal directions of the drawing surface). These observations about the characteristics of sketched components and their interrelations are relevant, because they provide the basis for using sketching as a form of human-computer interaction in GIS.

1.5 Intended Audience

This thesis is intended for researchers and developers interested in the design of future geographic information systems and in particular sketch-based query systems. Its intended audience includes researchers concerned with alternative, multi-modal forms of human-computer interaction, cognitive scientists interested in people's sketching behavior, computer scientists and database specialists looking for ways to assess the similarity between spatial scenes, and researchers concerned with formalizations and models of geographic space. The developed environment provides a test bed that may be of interest to scientists who want to perform experimental human subject tests or experiments with newly developed spatial formalisms.

1.6 Organization of Thesis

This thesis is divided into sections according to the four research questions postulated in Section 1.2.4. A chapter is devoted to each section and each chapter builds on observations and findings of previous chapters. The assessment of previous research, the evaluation of the hypothesis, and the conclusions are each in separate chapters. The reminder of this thesis is organized as follows:

The second chapter creates the link between this thesis and previous research by investigating different approaches to visual information retrieval. For this purpose a number of field-based and object-based information retrieval system are analyzed and compared. The chapter describes individual retrieval systems and discusses their strong and weak points.

The third chapter investigates current and alternative methods of user interaction. The main focus is on user interaction in GIS. We examine and discuss possible input and output channels of typical computer systems, especially those of sketch-based system. Conventional and alternative user modalities are evaluated regarding their applicability for specific tasks. User actions are investigated separately. This analysis is used to outline specific guidelines concerning the usefulness of alternative modalities, in particular sketching, within the scope of future GISs.

The fourth chapter studies the sketching behavior of people. For this purpose a survey was conducted, asking human subjects to draw sketches according to a set of written scenarios. These sketches are manually analyzed and classified. The interpretation of the survey focuses on object characteristics and spatial relations between sketched objects. The second part of the examination is concerned with binary spatial relations between sketched objects, focusing on qualitative and quantitative aspects. The results of this study are statistically analyzed and compiled into statements about typical characteristics of sketches and their basic components.

The fifth chapter develops the *digital sketch*, a model that captures essential characteristics of a sketch so that these can be processed computationally. The concept of the digital sketch is based on findings from Chapter 3 and 4. An association graph, consisting of nodes and edges that represent sketched objects and their binary spatial relations, is introduced. It is complemented by a method that allows reducing the number of edges (spatial relations) to a small set of relevant edges. The chapter investigates also the increase of size of the reduced and the complete association graph when objects (nodes) are added to the graph structures.

The sixth chapter discusses the prototype implementation of the sketch-based query system. The implementation is used as a test bed for evaluating the hypothesis (Chapter 7) and the developed concepts and formalisms. We review the internal model of the implementation, various aspects of the user interface, the processing sequence of a sketch, and how a sketched query is processed.

The seventh chapter evaluates the reduced association graph vs. the complete association graph. Various statistical methods are used to provide evidence for the support of the hypothesis of this thesis. The prototype implementation is used to query five different datasets containing sketches. Each dataset is queried once using a digital sketch based on the complete set of binary relations and once using a digital sketch based on the reduced set of binary relations. The results of both approaches are compared and interpreted qualitatively and quantitatively.

The eighth chapter concludes this thesis. Besides a summary of the thesis, it discusses the results of this research, as well as its implications considering the use of sketching to retrieve spatial information. The chapter also provides a description of future research activities enabled through this research. This outlook focuses on conceptual enhancements of Spatial-Query-by-Sketch, possible extensions of the prototype implementation, and research activities for which the current prototype application can serve as a test bed. The thesis closes with a portrayal of an integrated multimedia system that includes sketches as a valid data type.

Chapter 2

Visual Information Retrieval Systems

Information retrieval systems are based on methods that allow people to formulate requests in order to retrieve information. Visual information retrieval systems stress the use of visual tools to formulate a query. This approach is different from conventional information retrieval systems that use text-based query formulations, such as the *Structured Query Language (SQL)* (Chamberlin *et al.* 1976). Systems that are based on SQL and relational databases work well within application domains where data can be easily stored in tables. More complex systems that are used to manage and manipulate non-standard data, such as images, maps, or other multi-dimensional data, are often difficult to query with SQL or similar text-based query languages (Egenhofer 1992).

Visual query statements are frequently formulated as an example of a user's query. Systems using visual approaches focus, therefore, more directly on the *end result.* Text-based systems, on the other hand, often put an emphasis on the method that is required to retrieve information. Visually oriented systems require less knowledge about the mechanics of querying a database. It is likely that a visual query can represent people's mental worlds better than a purely textual expression. These advantages of visual over text-based query languages have led to the development of a host of new approaches, improving access to spatial data (Meyer 1993; Egenhofer 1996a). This development has also led to the insight that visual methods are best suited to query spatial information

(Calcinelli and Mainguenaud 1994). Visual information retrieval systems can be divided into those systems that allow a query formulation focusing on *fields* and those that focus on *objects*. The reason why some systems favor one approach over the other is most often linked to the type of data that is queried.

Fields are regions of space characterized by certain properties that can be determined at every point in the region. Individual points may have multiple attributes associated, but they store no explicit information about their environment or neighborhood. Images or digital terrain models (DTM) are typical examples of fields. A specific property of fields is that they are continuous within their boundary (Goodchild 1987).

Objects, on the other hand, are physical representations of entities that frequently represent higher-level information. They are typically created through refinement and interpretation of data. Objects have a unique identity and they can have attributes and characteristics, but also a specific functionality. Thus, objects can store knowledge about their environment (Laurini and Thompson 1992).

If an information retrieval system allows the user to query multiple characteristics (fields) or instances (objects) in the same query statement, then their relationships can be taken into account as well. This additional characteristic leads to a classification of visual information retrieval systems into four different categories:

- *Visual queries about the existence of fields*
- *Visual queries about fields and spatial relations*
- *Visual queries about the existence of objects*
- *Visual queries about objects and spatial relations*

The following sections review relevant visual information retrieval systems that fall into these categories.

2.1 Field-Based Retrieval

Systems that rely on field-based approaches to query information allow a user to specify local characteristics of a field. Such a characteristic can consist of a common attribute (e.g., red) or a specific pattern (e.g., a peak in a DTM) and its location can be fixed or it can be open.

2.1.1 Existence of Fields

These systems are primarily concerned with the existence of local characteristics. If multiple characteristics are queried then their interrelationship is not taken into account. However, some systems allow a user to explicitly specify the location of characteristics, in which case a relation to the reference frame–most often an image–is established.

TRADEMARK and ART MUSEUM

The TRADEMARK system was designated to retrieve graphical symbols, while ART MUSEUM was intended to search paintings in a large image database (Hirata and Kato 1992; Hirata and Kato 1993). The ART MUSEUM allows a user to draw a rough sketch of an overall composition of a painting (Figure 2.1). Alternatively a photograph or any other pictorial representation that is similar to the painting in question can be used for the same purpose. All images in the database must be preprocessed. The system extracts edges and generates metadata consisting of representative lines in a bitmap format for this purpose.

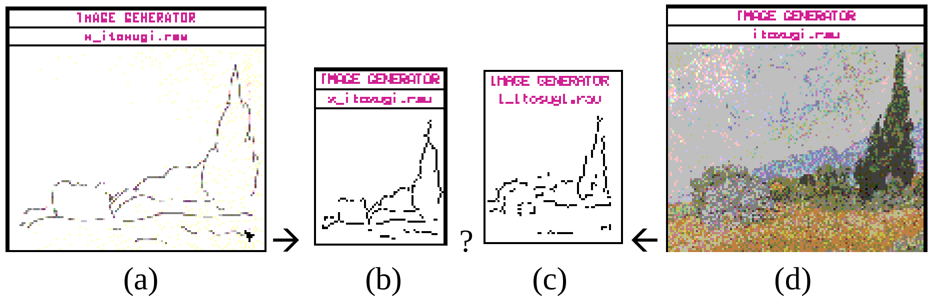


Figure 2.1 The basic principle of ART MUSEUM: (a) the original sketch and (b) the resulting line extraction that is compared with (c) the metadata extracted from (d) the original painting (Kato 1996).

The user's sketch is treated the same way as the images in the database: the sketch is thinned and brought into the same format as the image metadata. After this step, the two representations are compared with each other and a similarity coefficient is determined. The ART MUSEUM takes into account that outlines in a sketch may be partly incorrect (concerning shape and location). For this purpose the system decomposes the sketch into small square blocks that are shifted around to find the best match. The system also detects and controls "white or empty spaces" of a sketch, which may indicate that the user cannot remember the scene or that the space is either uniform or empty. Using the ART MUSEUM leads to good results if the user can specify the rough outlines of an image. However, even a moderate change of the location of sketched objects (e.g., moving the group of trees in to the middle of the sketch in Figure 2.1) makes a search less successful.

Fast Multi-Resolution Image Query

Another application with a focus on the retrieval of images from a large image database was presented by Jacobs *et al.* (1995). Here, a query is expressed by either a low-resolution image from a scanner or video camera or by a simple painting of a user. This approach explores a strategy based on wavelet decomposition of the entire scene of both the query and the database images (Chui 1992; DeVore *et al.* 1992). A transformation of an image into wavelets works much the same as simplifying a mathematical term by keeping only the most relevant components of the expression, for instance, the first few expressions in a Fourier row or the higher order expression of a multi-dimensional polynomial function. Depending on the chosen resolution only few arguments must be compared to get a representative statement about the similarity (Figure 2.2).

(a) (b) (c) (d)

Figure 2.2 The transformation of (a) the original image into (b-d) wavelets with varying resolutions (Jacobs *et al.* 1995).

Images and the query are decomposed and treated the same way. Each representation is considered as one single object with prominent areas. Because image and query are only compared by visual attributes, such as color and their absolute location, it is difficult to make a statement about the actual content of either the query or the image. The wavelet decomposition is appropriate for applications where large homogenous objects with specific patterns are searched–an example is the query of an outline of a lake with a particular shape.–but the approach fails if attributes of fields in the query vary from those in the image to retrieve (e.g., the comparison of two orthophoto of the same region with a lake may be unsuccessful, because the position of the sun can change the color of the lake).

Comparison Algorithm for Navigating Digital Image Databases

CANDID, the Comparison Algorithm for Navigating Digital Image Databases (Kelly *et al.* 1995) uses a query-by-example methodology to query large digital image databases based on an example image provided by the user. It generates an electronic fingerprint for each document in the database. These signatures are derived from various image features, such as texture, shape, or color information. A comparison of signatures in the database with the signature of the query example is used to assess similarity values and retrieve most likely matches. Like with other field-based approaches, CANDID works well if a user has good *a priori* knowledge about the images he or she is looking for. This includes all specifiable attributes, such as color or shape of fields, and their approximate location; however, if this initial knowledge is sparse, then it becomes more difficult to retrieve the desired information.

2.1.2 Fields and Spatial Relations

Some field-based systems that allow a user to query multiple fields at the same time can take into account the spatial relations between fields. For this purpose fields are considered regions or quasi-objects so that they can be spatially referenced to each other. Spatial relations between fields are typically defined based on the relative location of their centroids or Minimum Bounding Rectangles.

Query by Image Content

QBIC, Query by Image Content (Flickner *et al.* 1995) allows a user to search for images with a specific qualitative content in a large database. QBIC is based on an approach similar to that in the ART MUSEUM. The system processes color type and layout, along with texture, shape, size, orientation, and the position of connected regions (fields) in images. These fields have no attached meanings. Unlike the ART MUSEUM, QBIC allows the user to specify simple relations between fields. These relations are based on the relative position of centroids of outlined areas.

The query is generated either conventionally through a selection of supported properties from a table or graphically with an interface that looks like a simple paint-program where a user can draw and arrange objects or object-shapes. Querying for images with a specific average color or texture is done with a color-picker tool. The indexing of images is implemented by a multi-dimensional vector, using information about features, such as average color, color histograms, texture, shape, and position of detected objects. QBIC is intended primarily to retrieve photos, videos, or images that have a similar color composition as the query example (Barber *et al.* 1994).

Image Query by Semantical Color Content

Corridoni *et al.* have proposed a visual query method similar to QBIC to capture image properties, such as color quality and color arrangement (Corridoni *et al.* 1996a; Corridoni *et al.* 1996b). An image is represented by a set of icons that can be freely arranged in the query window. Each icon represents a local portion of the image, containing specific color related image properties. These icons focus on human perception of image attributes, such as color, hue, luminance, saturation, and warmth of a color. The key method in finding similar images is to compare local color histograms that are associated to every icon and their distribution over the query window in both the query and the images in the database. The implementation is specialized for the retrieval of art images and photographs, and in particular for imagery with large homogenous areas that stand in a particular spatial relationship to each other.

2.2 Object-Based Retrieval

Object-based systems focus on distinct entities rather than on characteristics of fields. Such systems allow a user, therefore, to query a database for the existence of objects with particular characteristics according to a user's query. Spatial relations can be considered when a query contains more than one object.

2.2.1 Existence of Spatial Objects

These systems are concerned with the existence of objects, while considering additionally certain object attributes, such as shape or geometry. This is similar to the equivalent approach of field-based systems. If multiple objects are queried, then their spatial relationship is not taken into account.

Mehrotra and Gary (1993; 1995) explore a method to retrieve similar shapes from an image database. Object shapes are extracted from each image and normalized, in order to make the inner representation invariant to scale, rotation, and translation. Each object is represented by a multi-dimensional vector of dimension $2 \cdot (n-2)$, where n is the number of vector points of the shape. The resulting vector consists of the x and y coordinates of the normalized points. Any pair of adjacent points of the extracted object shape can be used to form the normalized base vector (e.g., A’-B’ in Figure 2.3b).

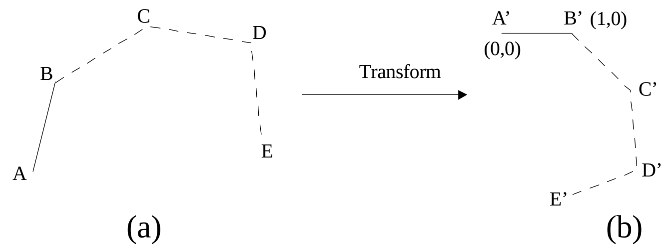


Figure 2.3 Example of a normalization of an extracted shape polygon (Mehrotra and Gary 1993).

In order to retrieve a specific shape, the user has to generate a query image that is treated the same way as the images in the database. The resulting multi-dimensional vector of the query image is then compared with the vectors of the images that have been previously extracted from the database. The similarity of any two given features is evaluated by the Euclidean distance between corresponding pairs of points in the multi-dimensional space. The method can detect shapes even if they are partially covered by other objects or if they are partially incomplete. The problem of distortion was solved to a limited degree by adjusting the tolerance on similarity measures in the search and by eliminating points with a vertex close to 180°. Retrieved shapes are not bound at any location and they can differ in size and orientation with the query. This method is, therefore, well suited to search an image object database for similar shapes; however, other object attributes are not considered.

2.2.2 Objects and Spatial Relations

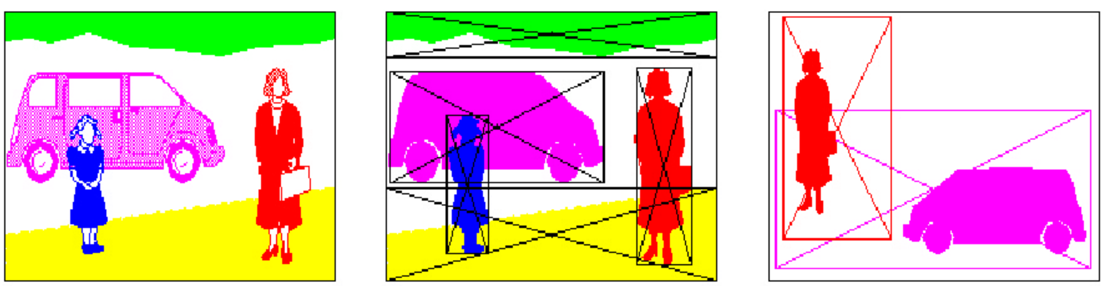
Systems that provide means to query objects and their spatial relationship provide the highest degree of flexibility to query spatial information. Two types can be found: Systems that use symbolic representations and systems that use sketches to formulate a query.

2.2.2.1 Icons and Symbols

These systems allow a user to formulate a query by arranging icons or symbols representing objects. Some systems allow a user to explicitly specify spatial relations between objects.

VisualSEEk and SaFe

VisualSEEk (Smith and Chang 1997) and *SaFe* (Smith and Chang 1999) represent advanced approaches to retrieving images. They allow a user to search for images by comparing local and global features, considering basic image properties and simple spatial relationships between homogenous image regions. The extraction of regions is based on color sets and on the principle of back-projection (Smith and Chang 1996). A query image is decomposed into different regions that are defined as areas with similar color and texture characteristics (Figure 2.4) and that form the basis for searching images that contain similar patterns (size and color).



(a) (b) (c)

Figure 2.4 Image decomposition into localized color regions: (a) the original target image, (b) the decomposed target image, and (c) a retrieved image with alternate positions of the query regions (Smith and Chang 1996).

Subsequent processing steps take into account only this subset of potential matches. All comparisons are based on MBRs, that is, the set of MBRs of the extracted regions in the query image are compared with corresponding information from images in the subset. This comparison is based on color sets and on the spatial distribution of the features' MBRs. The technique of spatial inference is based on 2D-strings (Chang *et al.* 1987). Since the absolute location of the query and target regions do not necessarily have to match, it becomes possible to retrieve multiple spatial object configurations of a scene (e.g., if a spatial scene is shifted). This allows a user to search more globally for images. However, objects are approximated with MBRs, this makes it difficult to specify an accurate spatial object configuration.

Cigales

Cigales (Mainguenaud and Portier 1990; Calcinelli and Mainguenaud 1994) is a visual and declarative query language especially developed for applications in GIS. It is based on the same query-by-example idea as VisualSEEk and SaFe, however, Cigales focuses on the retrieval of geographic scenes instead of images. A user can compose a spatial query with a limited set of predefined icons and relations provided by a graphical user interface. The user can create objects (regions and lines) to describe a spatial arrangement or to query specific objects (Figure 2.5).

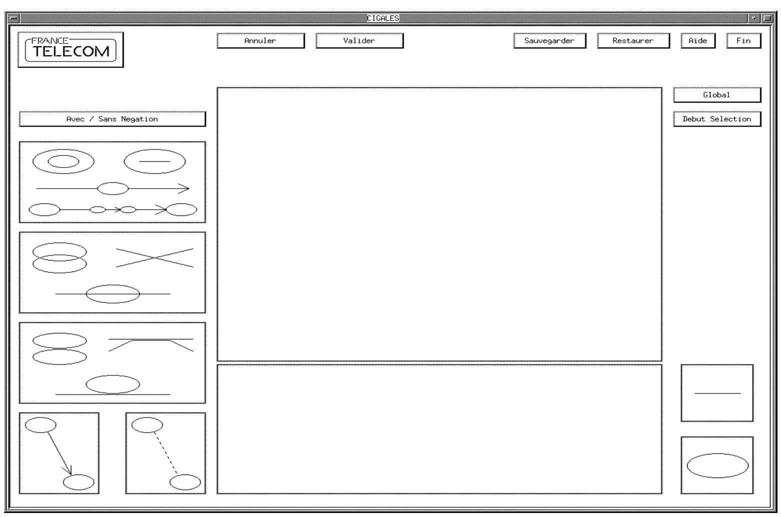


Figure 2.5 The Cigales user interface (Mainguenaud and Portier 1990).

Cigales was implemented with an extended relational formalism that uses a combination of several operators to define a query. Operators are defined as abstract data types and relationships are limited between a set of two objects. The set of operators is divided into user operators that are compatible with a user's mental model of spatial operators and system operators for the interaction with the extended relational database. This distinction has been introduced to simplify the handling of the system. The set of spatial relations on the user level includes intersection, inclusion, bordering, path, and Euclidian distance. A negation of these relations (except for the Euclidian distance) is expressed with a negation button. Spatial relations are transformed into extended SQL statements that can be performed against a spatial database. Strings (e.g., Townname = "Paris") and functions (e.g., Number_of_Inhabitants ≤ 10,000) can be assigned to objects as long as this type of information occurs in the database schema. The set of available spatial relations allows querying basic situations; however, Cigales lacks some important spatial descriptors, such as orientation, direction, scale, or closeness that are frequently used to characterize spatial scenes (Zubin 1989; Mark and Egenhofer 1994; Maaß 1995). An extension for these operators would increase the clumsiness of the interface.

3D Icon-Based Image Query

Del Bimbo *et al.* (1992; 1993) proposed an approach to query images independent of the point of view and in the third dimension. Their query language is based on a technique that represents objects as 3D-icons (Figure 2.6). Icons are three-dimensional MBRs. They are projected onto the three planes of an x-y-z coordinate system. With this assumption it is possible to specify spatial relations between any two icons, such as includes, coincidence, or is_included_with_left_adjacency. Spatial relations are established using 2D-strings (Chang *et al.* 1987).

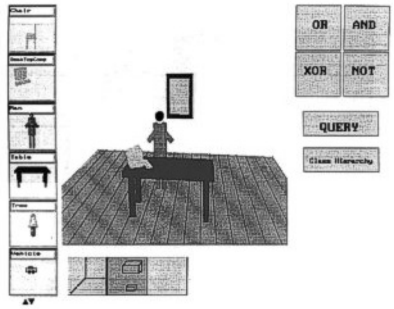


Figure 2.6 A sample of a query window with a scene composed with a set of icons (Del Bimbo *et al.* 1993).

Querying is based on the query-by-example philosophy. The user interface consists of a virtual three-dimensional room, where predefined 3D-icons can be placed, moved, and rotated according to the spatial configuration of the searched scene. An operator interacts with the system using 3D-gloves. After the interpretation of the spatial composition by the system, the 3D-scene is translated into a database query language (Object SQL). The query language statement includes parsed spatial relationships, object constraints, and logical expressions. The database stores images of 3D-scenes and a manually added, symbolic description of each image. This symbolic description is also the representation that is analyzed during the query processing.

Pictorial Query Language

The Pictorial Query Language, PCL (Di Loreto *et al.* 1995; Di Loreto *et al.* 1996) is a visual query language for geographic information that is based on point set theory and an object-oriented database model. Each object is defined either as a region, line, or point. The user formulates a query by selecting symbolic representations of objects from templates. The set of symbols includes basic object forms, such as squares or circles. Attributes and the type of objects are assigned separately. A small set of operators is used to describe the relations between objects. Besides the usual database operators there are operators for disjunction, adjacency, inclusion, equality, overlapping, and crossing. A distance operator, consisting of two components, is used for metric considerations. The first component is used to express a minimum/maximum distance relationship between the two objects. The second component can include conventional metric operators, such as *less than*, *equal*, or *greater than*, which can be applied to the first distance component.

Vivid Spatial Constellation (VISCO)

The Vivid Spatial Constellation, *VISCO* (Haarslev 1996; Haarslev and Möller 1997) relies on the principles of deduction and description logic (Haarslev and Wessel 1997). The system allows the integration of topological and metric properties into a spatial query. VISCO uses various metaphors for the description of points (nails and marbles), line-segments, and poly-lines (rubber band, beam). Each spatial scene is drawn on a transparent film that can be skewed or rotated. Metric is introduced by using enclosures, which are equidistant regions on the interior or exterior of sketched objects. Queries rely on a combination of topological and metric constraints. A relaxation of a spatial scene using "don't care" conditions is visualized by animating objects. VISCO incorporates a set of powerful tools to formulate spatial queries; however, the handling of different information layers combined with an abstract symbology might prove difficult for casual users.

Icon Query Language for Topological Relations in GIS

Lee and Chin (1995) propose an approach similar to PCL with a user interface that consists of four parts: (1) an icon template, (2) the space to formulate an iconic query, (3) a map display window, and (4) a text output/verification window. The Icon Query Language allows a user to specify a spatial query by drawing icons consisting of rectangles, lines, and points. Lee and Chin use a set of predefined operators to describe the relationships between drawn icons. The set of operators is different for each object type. Line-region relationships feature operators, such as *enter, exit*, or *through*, in addition to commonly used operators, such as *inside* or *outside*. Metric constraints are specified using dialog boxes and menus. A query verification window monitors the parsing and interpretation process of the system. Should the application make an interpretation error, it is up to the user to correct the problem by either removing the constraint or removing the icon in this window. Because a query formulation is based on geometric primitives, more complex queries become quickly difficult to comprehend, making the query building process *per se* dependable on the solid abstraction capabilities of a user.

2.2.2.2 Sketch

Sketching is a less constrained method to formulate a query than composing a query using icons or symbols. Sketch-based systems allow users, therefore, to express themselves more freely.

Electronic Cocktail Napkin

The *Electronic Cocktail Napkin* (Gross 1994b; Gross 1994a; Citrin and Gross 1996; Gross 1996) aims at helping designers and engineers to bring their ideas on electronic paper. It can recognize, interpret, and manage sketched glyphs, gestures, and diagrams. The application incorporates constraint, shape, and character recognition features and recognizes hand drawn input in a three-step process. The lowest level of interpretation recognizes multi-stroke symbols, glyphs, and gestures. The system provides gestures for operations, such as to *erase*, *copy*, or *pick* objects (Figure 2.7). These gestures can be trained and they are recognized much like handwritten characters.

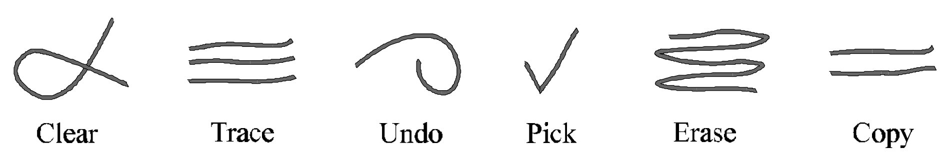


Figure 2.7 The set of gestural commands offered by the Electronic Cocktail Napkin (Gross 1994a).

The second level of the parsing process analyzes spatial relations among the detected symbols or objects, and the final step matches detected objects and relations with known configurations from a set of templates. The spatial interpretation module can specify natural language predicates for binary spatial relations, such as *above*, *immediately above*, or *right of*. However, this interpretation is imprecise, because relations are based on MBRs, and start or end points of strokes. Additional recognizers can detect structures, such as links between circles, or names in boxes in a hierarchical manner. The application does not automatically substitute drawn glyphs with geometric shapes, that is, both representations are stored and the user can decide which representation is to be used. To achieve good results, the Electronic Cocktail Napkin has to be trained with each individual user. While the multi-level interpretation and the modularity of the application are of great interest, the Electronic Cocktail Napkin lacks a formal base for the representation of a sketch.

Image Retrieval by Sketch

Del Bimbo and Pala (1994; 1997) describe a system that allows a user to retrieve objects based on sketches. The sketched shape of an object is transformed into an elastic template that is matched with extracted object shapes from images. The similarity of two shapes is calculated as a function of the local stretching and bending factor that is necessary to bring the elastic template shape into the extracted object shape of the image and an overlapping factor (Figure 2.8). The algorithm is invariant under translation, but not under rotation and scaling. A change in scale between template and extracted object shape results in increased deformation energy and, therefore, in a score with a lower similarity value.

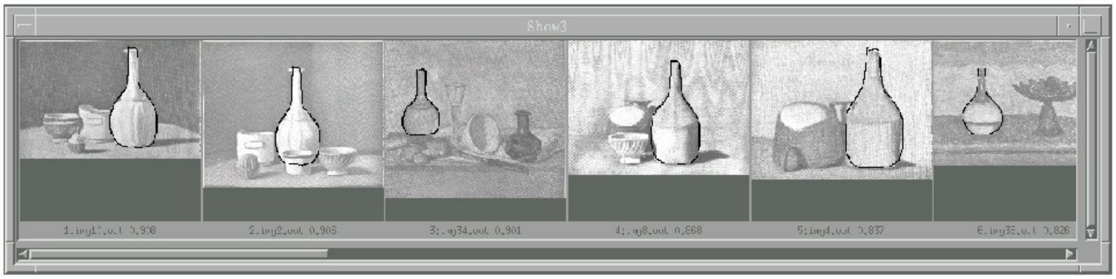


Figure 2.8 Detected bottle-like shapes from a library of paintings that have been retrieved based on a freehand sketch of a bottle (not shown on this figure) (Del Bimbo and Pala 1997).

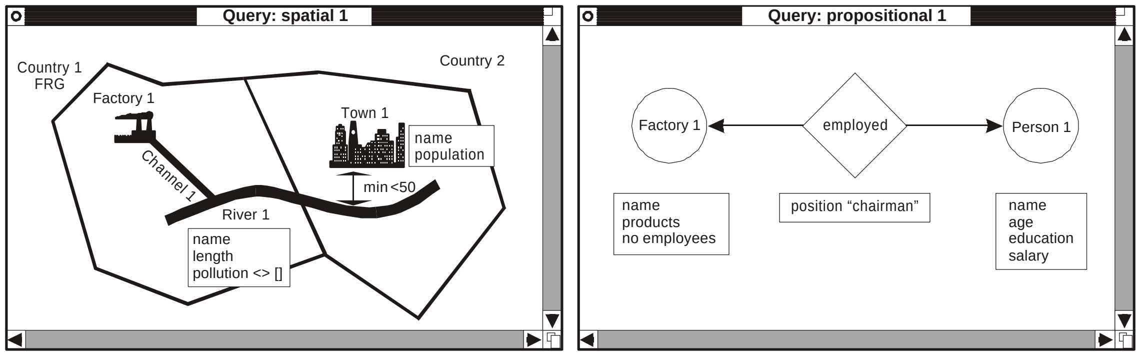
Spatial relations between objects are taken into account by comparing the objects' projections on the x and y-axis (Chang *et al.* 1987). The result is one of five possible spatial relationships for each axis: *disjoint*, *meet*, *contain*, *inside*, and *overlap*. To reflect the two-dimensional notion of this representation, Del Bimbo and Pala introduce four additional orientation parameters that capture the directional relationship between two objects. To speed up the query and matching process, images in the database are preprocessed and a signature file with information about objects and relations is maintained.

Image Query by Sketch

Image Query by Sketch, *IQ* (Agouris *et al.* 1999) is tailored to the retrieval of images. The query formulation is based on a sketch of a spatial scene consisting of one or more objects. Shape and spatial configuration of sketched objects are taken into account. The objective of the query is to retrieve a set of images that contain spatial object configurations that are similar to the sketched query. The system is based on a database with three libraries. An *image library* contains the unprocessed digital images. The *metadata library* maintains lists with attributes and other descriptors that were previously extracted during the preprocessing of the image library. The third library, a *feature library*, stores distinct features, such as shapes, and links them to images in the image library whenever such features have been detected. This approach helps to make subsequent queries more efficient, because only the feature and metadata library need to be consulted. The matching procedure is based on a variation of the least square matching (LSM) method (Agouris and Schenk 1996). The spatial location of potential objects is expressed by their MBRs. The evaluation of the spatial similarity between a sketched query and the spatial scene in a candidate image is based on an analysis of these MBRs. For this purpose IQ considers topological characteristics of spatial relations between objects according to the 9-intersection model (Egenhofer and Herring 1990; Egenhofer and Herring 1991).

Sketch!

Sketch! (Meyer 1993; Meyer 1994) is a query language for spatial information systems, using visual-logic. A spatial query has an analogue and a propositional part. The term *analogue* stands for a statement of quantity, measure, or distance in a scene, whereas *propositional* refers to a description of an object's existence and its properties. One example for an analog feature of an object is its shape or its position, whereas its color would be propositional. The user interface consists of two types of windows (Figure 2.9): Propositional information is displayed in *p-windows* and analogue information in *s-windows.* Non-spatial information is only displayed in *p-windows*. Objects are represented as circles, links as diamonds, and associated attributes are stored in tables next to objects or links. The spatial configuration of objects is composed in *s-windows*. For complex spatial relationships, it is possible to extend the scene over multiple p- and s-windows.



(a) (b)

Figure 2.9 The dual window user interface of Sketch! with (a) the analogue *s-window* and (b) the propositional *p-window* (Meyer 1993).

The set of permissible objects includes points and open or closed lines (areas). Only objects that occur in the database can be used to formulate a query. To create a new object, the user has to drag an icon from a template onto the drawing window. The type of object has to be chosen from a pop-up menu prior to drawing. This procedure has to be repeated for every newly generated object on the scene. An object is described solely by its position and its extent. Other spatial object properties, such as shape, closeness, or orientation, are not implemented. Objects can be linked using such spatial predicates as *closest*, *intersects*, *neighboring*, *inside*, and *touches*. The set of spatial functions includes *min_dist*, *max_dist*, *intersection*, *common_border*, and *center*. Relations that are non-spatial or invisible in the s-window must be defined in the p-window by dragging objects from the s-window into the p-window and specifying the desired relations. A sub-scene can be represented as an object itself.

The complexity of interaction increases rapidly when multiple objects with inter-linked relations are involved. Due to the dual window approach a user has to trace objects in at least two windows. If these objects have relations to many other objects they will reoccur in all sub-windows as well. Hence, a user has to be aware of all relationships and objects in various *p-* and *s-windows* pairs. The redundant object representation and the breaking of spatial scenes into small sub-scenes can lead to complicated and confusing queries.

Spatial-Query-by-Sketch

Spatial-Query-by-Sketch, SQbS (Egenhofer 1996b) is the framework for this thesis. It is an advanced concept of a sketch-based user interface to query spatial information within a GIS. SQbS is based on a multi-modal paradigm of user-computer interaction that allows a user to query a spatial database by drawing freehand sketches.

Similar to the Electronic Cocktail Napkin, SQbS associates the term of *sketching* with freehand drawing rather than with the construction of geometric figures or with the composition of icons or symbols. By choosing this approach, we try to come as close as possible to the original meaning of the electronic paper metaphor (Kuhn and Frank 1991; Kuhn 1992; Kuhn 1993). SQbS provides of a spatial logic that detects and corrects inconsistencies between drawn objects and that adjusts imprecise or incompletely drawn objects automatically. The system generates a *digital sketch*, once the sketched scene is complete and all conflicts are resolved. This digital sketch is an object-oriented footprint of the sketched scene that describes objects and binary spatial relations between them. The digital sketch is the base for the spatial query.

The scene similarity between the sketched query and an individual spatial scene in the database is computed based on the similarity of topology, metric, and direction of binary spatial relations and the geometric similarity between objects. A spatial query is automatically relaxed if the system has not retrieved similar scenes. The result of a query is presented in a visual form. Relevant visual objects of the result can be hyper-linked so that the user is able to obtain additional information if this is necessary (Benderson *et al.* 1996). New database queries can be formulated on top of retrieved results. This allows a user to dynamically and incrementally query spatial information (Williamson and Shneiderman 1992).

Spatial-Query-by-Sketch is based on a *comprehensive formal model* of spatial relations that is not matched by any of the previously reviewed spatial query systems. SQbS is also the only *pure sketching* interface, allowing a user to express his or her mental model of a spatial scene in a direct manner. Thus, SQbS is a *natural* and *intuitive* tool that keeps the query simple, by avoiding detours over icons, graphs, or geometrically constructed objects. SQbS relies on people's basic sketching skills and the query process avoids multiple layers and other learning intensive mechanisms that can distract a user from the task to accomplish. Hence, and in contrary to other approaches, the communication effort is delegated from user to the computer.

2.3 Summary

Visual information retrieval systems allow people to access information using visual query tools, which is different from many contemporary information retrieval systems that are based on text-based query mechanisms. Most visual queries are based on the *query by example* metaphor, that is, the database is searched for information similar to the user's query. Depending on the type of information to be queried two different conceptual approaches of visual information retrieval systems have been developed: (1) those systems that focus on retrieving *fields* and (2) those that focus on *objects*. A further distinction can be made if a system provides a mechanism to take into account the spatial relationships between fields or objects. This chapter reviewed systems from all four categories, discussing their advantages and disadvantages. Field-based systems are suitable for applications, such as retrieving images, paintings, or other raw and unstructured data types, whereas object-based approaches are appropriate to query information that is stored on a more abstract and refined level. Taking additionally into account the spatial relations between fields or objects increases the expressiveness of a query. Many applications and operations in GIS focus on geographic objects and their interrelation. The object-relation-based concept is, therefore, the method of choice for querying spatial information.

Chapter 3

Design Guidelines for a Sketch-Based GIS User-Interface

The virtual and physical area of interaction between user and computer is called the *user interface* (Encyclopedia Britannica 1996). Early computing applications could not afford to spend a lot of resources for sophisticated user interfaces, because the available computing power was used primarily for processing tasks. Today's desktop computers have enough CPU power to run computation-intensive user applications *and* to sustain a sophisticated user interface.

The design of a user interface involves many issues, beginning from hardware devices, such as mouse, keyboard, monitor, or a pen, over application specific interaction procedures to psychological and cognitive characteristics of people. An interaction works best if all components and processes are mutually adapted to each other. This implies that the user feels comfortable and the sequence of events is intuitive to the user (Gould and Lewid 1985). A typical user operation consists of a sequence of user actions that must be synchronized with computational processes, such as output generation, input verification, and other process-related tasks. Mastering this complex system of person-machine interaction is challenging and requires a careful design (Chignell and Hancock 1988). This chapter compares various methods to interact with computer systems, emphasizing spatial information retrieval systems and sketching. The resulting considerations are summarized as *design guidelines* for visual user interfaces.

3.1 User Modalities

Modalities address any type of sensation, including vision, hearing, and various ways of expression, such as writing, talking, or gestures that people use to interact with one another (Neal *et al.* 1988; Wexelblat 1995). A multi-modal interaction involves more than one form of communication. An example of a multi-modal interaction between two persons is if somebody draws a sketch and explains at the same time his or her sketch verbally. A multi-modal user interface is a user interface that offers multiple concurrent input and output channels. A channel in this context is a medium for an interaction between user and computer.

Multi-modal communication among people is quite common. In addition to speech there are less prominent human communication channels, such as eye contact, face-mimic, and other non-verbal forms of expression, such as body-signals or gestures that help people to exchange information (Wexelblat 1994). Multiple modalities are often used simultaneously to clarify or emphasize certain aspects of communication. At the same time, this redundancy may result in contradictions and, therefore, the exchange of information may become unclear and ambiguous. The effectiveness and performance of an interaction depends on the characteristics of the chosen modalities, their synchronization, and on the capabilities of the information sender *and* receiver. What modalities are chosen in a specific situation depends on various factors, including the task to accomplish, personal skills, mood, and the availability of modalities (Kuhn 1992).

An interaction with a computer is quite limited if compared with the rich forms of inter-human communication (Buxton 1986). Most of today's systems allow users only to point and type and they produce—aside from electromagnetic waves and sounds—only text and graphics on the computer screen. This situation could be improved if the field of possible modalities would be enhanced with additional visual or acoustic communication techniques, such as sketching and talking.

Figure 3.1 shows the set of potential user modalities and typical user actions for a sketch-based user interface. Four modalities are suitable to communicate *to the system*—

pointing, writing, sketching, and talking—and two modalities are used to perceive information *from the system*—seeing and hearing. The set of provided user actions (select, compose, operate, change view, and perceive) covers a typical interaction between user and computer.

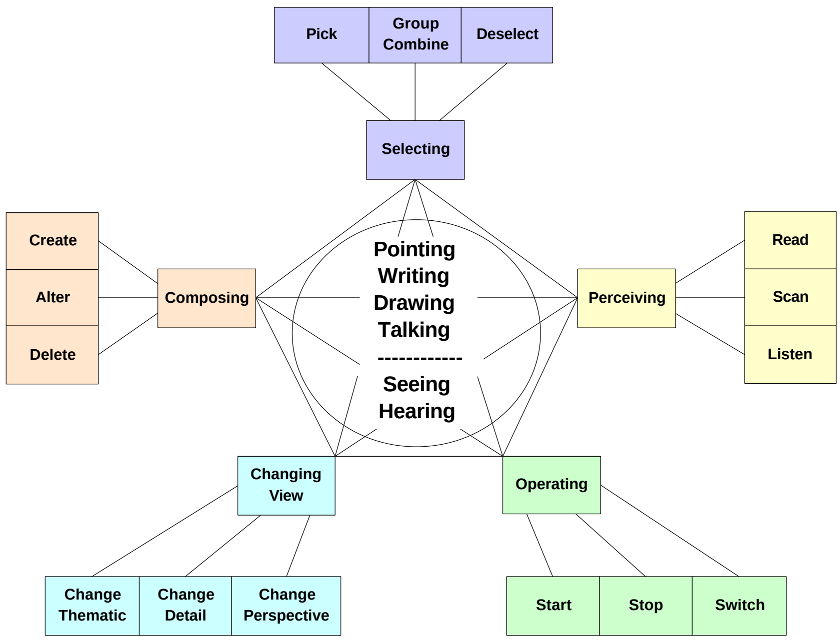
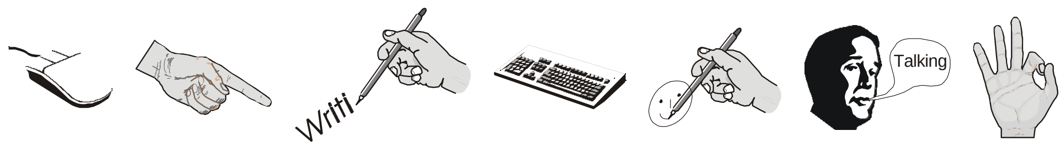


Figure 3.1 The set of potential user modalities and typical user actions for a sketch-based user interface.

The following sections analyze this set of user modalities according to their suitability for typical user tasks. The result of these considerations is a number of recommendations that highlight advantages of individual modalities. These guidelines apply to people who have no deficiencies with respect to modalities. Universal access is concerned with HCI for everyone. In case of underdeveloped modalities, these must be compensated through modalities that are well developed.

3.1.1 People's Output Channels

The main communication repertory of people consists of *pointing*, *talking*, *sketching*, *gesturing*, *writing*, and *typing* (Figure 3.2). This sequence is consistent with the temporal order in which people learn these modalities in their childhood (Owens 1996). The level of difficulty increases also in the same way. Visual interaction, such as pointing and sketching, are universally understood gestures, while verbal interaction, such as talking and writing, are more closely tied to a certain culture.



(a) (b) (c) (d) (e) (f) (g)

Figure 3.2 The set of a user's output modalities for human-computer interaction: (a) indirect and (b) direct pointing, (c) writing, (d) typing, (e) sketching, (f) talking, and (g) gestures.

Other modalities, such as lip-reading or eye-tracking (Vo and Waibel 1993), are less applicable in the scope of a sketch-based user interface.

3.1.1.1 Pointing

Pointing is the most natural gesture. Small children start to communicate with their environment by pointing, which helps a child to formulate questions or express delight (Collins 1979). Pointing is used where other modalities fail if they are too sophisticated or too slow, or where there is no other mutually understood way of communication, such as telling a stranger the way to the train station. People point using their hands or fingers; however, it is also possible to point virtually by glancing in a particular direction or at a particular object. Pointing is a modality that focuses on one object or direction at a time.

Guideline 1: *Pointing is a suitable interaction modality to select visible objects, initiate processes, or to set the focus.*

3.1.1.2 Talking

Children normally start to talk between the age of one and two years (Levick 1986; Owens 1996). They assign names to things and activities, and they combine expressions, which builds the base for a powerful and sophisticated way of communication that goes much beyond pointing. Similar to pointing, no special tools are required for talking. Since words can be precise or vague, it is important to carefully consider their sequence and semantics, which can differ from one natural language to another. The receiver of spoken information must be synchronized with the sender, otherwise the content of the message is difficult to understand and pieces of information may get lost. Talking is, therefore, the modality of choice for short or rapid forms of information exchange that can be understood in a single approach. Consequently, talking is appropriate for short instructions or annotations, but less suitable for long and complex monologues.

Talking is a sequential modality, because spoken components must follow each other in a chronological order. Talking is well-suited for the description of sequential events, such as a football game. Conversely, the verbal description of a complex situation, such as a spatial scene on an image, can be difficult, because every relevant object and object relation must be explicitly explained to convey the complete picture (Allen 1995). This situation can be improved if talking is enhanced by visual means, for instance, when a verbally presented text is also visible to an audience on an overhead projector.

Guideline 2: Talking should be used primarily for applications that require short or rapid forms of interaction.

3.1.1.3 Sketching

About at the same time when children start to express themselves verbally they begin to communicate through drawings as well (Levick 1998). The first stage is sketching single objects (2-4 years). Later, children depict objects within a specific context, involving more detail (4-7 years). Drawing typically requires a tool, such as a pencil, ballpoint, or a brush, whereas pointing is less dependent on external devices. Similar to pointing, however, sketching is a visual form of expression and a universally understood modality that excels when complex situations must be expressed. Drawing is a truly two-dimensional form of expression. In more elaborated drawings also the third dimension can be expressed. A sketched composition of a situation is non-volatile and people can re-scan it if the intended message could not be perceived in the first place. Therefore, drawings are well suited for generalization, interpretation, and visualization purposes of complex spatial, conceptual, or hierarchical scenarios (Blaser *et al.* 2000). Another useful characteristic of drawings is that they can convey a lot of implicit, preprocessed information, because they are the product of perception, reflection, interpretation, and visualization.

Guideline 3: *Sketching is an appropriate interaction modality to express spatial, conceptual, or hierarchical configurations that are otherwise difficult to describe.*

3.1.1.4 Writing

Among the modalities reviewed, writing is the most difficult to learn (Owens 1996). Children typically begin to write when they go to kindergarten or first grade. Written language is built on a well-defined set of symbols, called characters. Sequences of characters make up words, which form sentences, when combined. Writing requires people to remember the syntax of words. The basic principles of spoken and written language are the same, namely the sequential flow of verbal statements, also referred to as sentential representation of information (Larkin and Simon 1987). The difference between both forms of communication is that while talking is *volatile*, writing implies that language is recorded in a persistent way so that the verbal statements become *non-volatile*. This visual component makes writing a *semi-visual* modality. An example is a person scanning the frontpage of a newspaper for keywords, a technique that is comparable to the process of quickly analyzing an image for important details. Writing is typically slower than talking; however, it allows people to construct more complex textual structures that are still understandable. People are also more careful when they write than when they talk, considering grammatical and syntactical rules. Written language is, therefore, typically more precise and expressive.

Today's methods of interaction with computer systems are based primarily on typing and pointing, while writing freehand text is used only for some niche applications, such as Personal Digital Assistants. The detection and interpretation of text entered in freehand style is still computationally expensive. Typed input, on the other hand, can be processed in a more efficient way, which is because the interface for typed input is narrow, consisting of a mere subset of the standard ASCII character set.

Guideline 4: *Freehand writing and typing are important modalities for entering text or for communicating precise verbal statements with a computer.*

3.1.2 People's Input Channels

The communication between user and computer is bi-directional, that is, people and computer are both *sender and receiver* of information. Channels with which people perceive information are equally important as human output channels. The two main input modalities are hearing and seeing (Figure 3.1). Other human sensations play a subordinate role in today's human-computer interaction.

3.1.2.1 Hearing

Hearing is a sequential modality. The fact that people can distinguish and tell sounds coming from different directions provides them with additional spatial information. Hence, while seeing is bound to a specific direction, the perception of sound is only bound to intensity. The human ear is excellent for separating different sounds from each other, but people become distracted when they have to listen to more than one acoustical source at the same time. The sensation of hearing cannot be turned off and people register noise subconsciously, even if they do not explicitly listen. Sounds play only a minor role in human-computer interaction.

Synthesized verbal communication (i.e., verbal statements that originate from a computer system) must be adapted to the user's pace of interaction (Oviatt 1999) and the quality and content of verbal communication must be high and natural. Computer devices that talk often distract the environment when they are used in public. Because the voice of a computer (its "mood") is always the same, spoken communication generated with a synthetic device becomes monotonous and boring. However, there are some applications where computer generated voices are used, notably in environments where users are focused on operating or navigating a device, such as an airplane or a car. Otherwise sound is primarily used for acoustic signals to attract the user's attention, whether it is to report an error or an event, such as to announce incoming mail.

Guideline 5: *Hearing is a suitable interaction modality to perceive simple unambiguous signals from a computer system.*

3.1.2.2 Seeing

Our modern society is extremely visually oriented and that is probably one reason why today's computer interaction is mainly based on the sensation of seeing (Buxton 1986). The maximal resolution of a human eye is very high, approximately 40 times higher in each cardinal direction than the resolution of today's output devices (i.e., a typical computer monitor). People do also well in perceiving colors, contrasts, and edges. In combination with the human brain and paired with a great knowledge about shape, texture, and other visual attributes of real-world objects, our eyes are very effective in detecting coherent structures, even if the perceived image is unclear and confusing. Compared with hearing, which is optimized for singular input, optical perception allows people to track multiple objects at the same time by rapidly scanning the visual range. The sensations of seeing and hearing are complementary and extremely effective when used together. Seeing can be used to track and control processes, to read messages, or to follow the cursor or other visual objects.

Guideline 6: *Visual perception is the primary input channel for human computer interaction.*

3.1.3 Modalities in a Sketch-Based User Interface

The main goal in user interface design is to create a well-balanced and mutually adapted working environment that takes into account the specific characteristics of the user and computer side (Mountford 1992). From the user's point of view this means that the set of user modalities must be simple, natural, and intuitive. In a sketch-based environment people can use a pen or their fingers in addition to conventional modalities to interact with a computer. A pen can be used for pointing, writing, and drawing. If talking is a permissible modality, then voice can be used for giving short instructions. Fingers are preferably used where precision and accuracy play a subordinate role. Probably the most important distinction between a sketch-based user interface and today's user interfaces is that the user can express spatial thoughts *directly* and that sketches are interpreted with respect to their *content*, instead of being stored as simple bitmaps. Hence, sketching and drawing fills in the gap where conventional methods and verbal descriptions fall short, for instance, when a user wants to communicate complex structured spatial, hierarchical, or conceptual information (Blaser *et al.* 2000). To keep the user interface flexible, it is important to leave the choice of modality to the user. The output of a sketching device will be mainly built on intelligent visual modalities, but supported by acoustic signals. *Intelligent* in this context stands for guiding, supporting, and providing only the most relevant and recent information, without distracting the user.

3.2 User Actions

The term *user action* stands for an elementary user-computer interaction that can be of an active or passive type. Active is connected with *doing* something, while passive is associated with *perceiving* something. User actions must not be confused with *user activities* that have been investigated on a practical level in (Cypher 1986). User activities address a set of consecutive individual actions, called *user operation* (Blaser

1997). We distinguish five distinct types of elementary user actions on which any human-computer interaction can be built: *perceiving*, *selecting*, *composing*, *changing view*, and *operating* (Figure 3.1).

Perceiving, selecting, and composing are primary activities, while changing view and operating are supporting activities. The efficiency and usability of a system can be measured by counting the number of elementary actions that are necessary to perform specific user operations (Card *et al.* 1980). In addition to this quantitative approach, it is possible to anticipate the quality of a system by considering the simplicity of individual user actions and operations.

Guideline 7: Effective concepts of user operations are based on a small number of easy and intuitive actions.

3.2.1 Perceiving

No interaction without perception. Good perception is essential for a well functioning computer interaction. While certain actions are mandatory, perception is not. People's abilities to perceive relies primarily on seeing and hearing, therefore, the three basic activities concerned with perception are reading, scanning, and listening (Figures 3.1 and 3.3).

Reading refers to the perception of sequential text. *Scanning* can be applied to text and graphics, focusing on the main spots of interest, such as titles, headings, or emphasized parts in graphics. *Listening* is concerned with the perception of sounds. An application can deploy various techniques to guide a user to perceive a specific message. One possibility of getting a user's attention visually is to create a local contrast in the user interface. Such a discontinuity can involve color, shape, or size of existing elements or new superimposed graphical elements. Moving objects or animated elements in an otherwise static display, such as a progress bar, provide other ways to attract a user's attention. The same holds true for the acoustic channel. However, acoustic information is less focused than visual information if acoustic signals are without visual or verbal underpinning. For instance, a beeping system that provides no additional clues about the source of the sound is insufficiently descriptive.

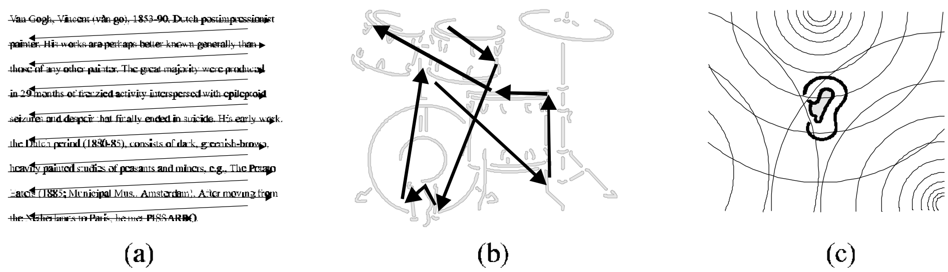


Figure 3.3 The three basic user actions considering perception: (a) reading, (b) scanning (here in the context of scanning a picture), and (c) listening.

Guideline 8: The visual channel is a system's principal channel to convey information in HCI. The acoustical channel is useful to attract a user's attention.

3.2.2 Selecting

Selecting is an essential form of human-computer interaction and usually the first user action during a user operation. Whether to manipulate an object or to initiate a process, selecting items is essential. Selection–the act of selecting–in early computer systems was primarily based on using a keyboard. To select an item from a list or a menu, the user had to press a key or a combination of keys. Many systems at that time were command-line based, what made them more difficult to operate (Egenhofer and Kuhn 1999) and the use of arrow keys for navigating and selecting was only a small progress. However, the situation improved dramatically with the introduction of graphical user interfaces (Kay and Goldberg 1977; Goldberg 1988), which made it possible to access and manipulate user interface elements directly (Shneiderman 1983; Shneiderman 1997). Selecting includes the following subtypes: picking, grouping, and deselecting (Figure 3.1).

All output modalities can be used for selecting; however, pointing appears to be the most intuitive technique, because it is simple, direct and, therefore, efficient. Drawing gestures, such as circling or underlining can also be used for a selection (Rubine 1991). Sketched gestures are an intuitive way to interact with a system, because they belong to the natural repertoire of people's ability to express themselves. One can, therefore, expect that inexperienced computer user will welcome such a simple form of interaction. A direct selection is limited to visible elements in the user interface.

Guideline 9: *Pointing and drawing gestures are the modalities of choice for selection. Simple, spoken natural language statements may be used simultaneously, either to support a visual selection or to resolve ambiguities.*

This combination of gestures and verbal communication renders the process of selection more efficient when multiple objects have to be addressed (Oviatt 1997) or more reliable, when the same object is specified with both methods (Neal and Shapiro 1988).

3.2.3 Composing

Composition, the combining of distinct parts or elements to form a whole, represents the *creative part* of human-computer interaction. In HCI, the primary objective of a composing is to create or manipulate textual or graphical objects. Based on the composition of basic objects, there are derived "objects", such as relations or impressions, that cannot exist by themselves. An impression in this context is the overall notion of a composition or a sub-structure of a composition that requires an interpretation effort. Impressions lack the tangibility of relations, because their nature is subjective. Composing usually follows selecting and involves one of the following actions: creating, altering, and deleting.

A *creation* starts when "something new" is initiated, for example, after selecting a drawing tool, when the first dot is drawn. New components can also emerge when previously created components are combined. Such a combination can be implicit or unintentional. Implicit objects can be of great relevance for the overall understanding of the creation. For instance, an intersection between two lines in a sketch (i.e., an implicitly created object) can be the most important object in the scene. Composition is an iterative process and, therefore, it must be possible to alter and delete created elements. An *alteration* is an action that changes an already existing element, while a *deletion* of an element stands for its discontinuation of being existent.

Guideline 10: All output modalities are acceptable for compositions. The suitability of a particular modality depends on the task to perform and the user's skills and preferences.

3.2.4 Changing Views

People are used to change their views of things depending on the actual task. Expressions such as *take a closer look at something* or *look at something from a different perspective* are evidence for this behavior (Benderson *et al.* 1996). Changing views is an essential aspect for many applications. For instance, if someone has a folded newspaper, he or she might be able to read the headings, but in order to get more detail, first the newspaper has to be unfolded (that is, the view has to be changed). The same applies when working with a computer interface. In order to work efficiently the user must be able to perform a change of view depending on the actual situation. There are three different concepts for changing a view: change of perspective, change of theme, and change of detail (Figure 3.1).

The *change of perspective* (Figure 3.4a) stands for actions such as panning, rotating the viewport, or changing the viewpoint of an object from the observers perspective (Benderson and Hollan 1994). A change of perspective must not be confused with an alteration of the geometry or spatial configuration of visible elements, because only the viewport is changed, while all objects remain untouched. This statement is valid for all view changes. A change of perspective can be applied to graphics and text. In a 2D-environment, such as a picture, a change of perspective may include actions, such as rotate, pan, or zoom. In a 3D-environment it could involve a user virtually to fly through a spatial scene (Feiner 1992). The concept of a view implies always *perception* and *interpretation* of a visual scene.

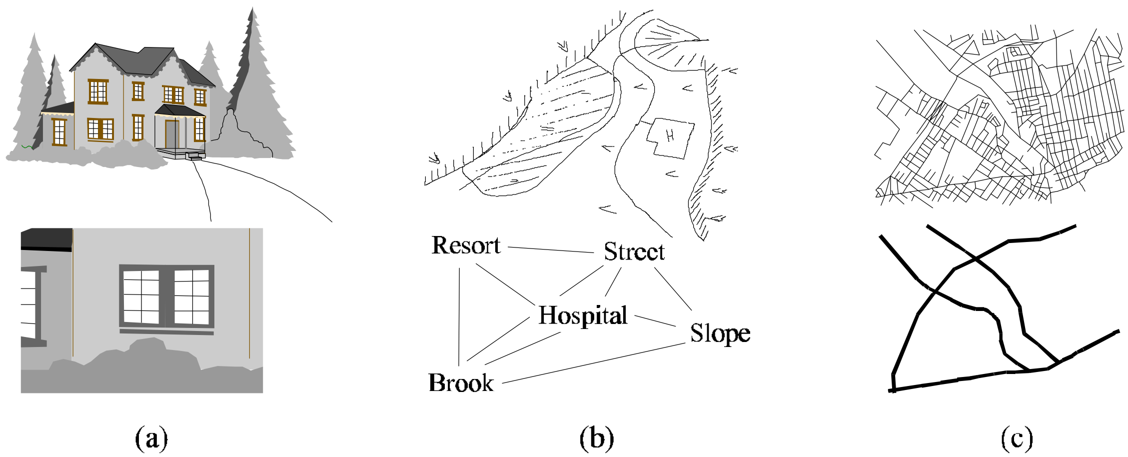


Figure 3.4 Three basic concepts for a change of view: (a) perspective, (b) thematic or conceptual level, and (c) detail.

Johnson (1995) compared user preferences for changing the viewport by panning. He found that most users prefer the method of moving the viewport by pushing the desktop (background) into the desired direction, instead of moving the frame of the viewport over the scene. For fast access to all parts of the real estate of a user interface it is beneficial to have a minimized icon of the entire desktop, like it is used in many UNIX X-Windows managers.

Changing the *thematic level* (Figure 3.4b) means looking at the same data but through different lenses or with a different filter (Stone *et al.* 1994). An overview or a different conceptual view, for instance, can be considered a thematic layer. A change of view is necessary when only selected features are of interest, such as when the density of information is too high or when the user wants to change the conceptual perspective (Stone *et al.* 1994; Timpf 1999). The classic example of an application working with thematic layers is a GIS (Maguire *et al.* 1991).

People can easily change the *level of detail* (Figure 3.4c) in their everyday life, for instance, simply by approaching or withdrawing from objects. However, changing the granularity within a user-computer interaction is more difficult and requires in general a considerable effort of the user interface (Jackson 1990). Typically, a change in detail is necessary when a user zooms in or out on information. Whether to drop or to reveal object details cannot simply be determined as a function of the actual geometrical size of an object. The decision what objects and properties to show and which ones to hide depends on the relevance of objects, the context, and the configuration of a scene. The overall goal is to keep the portrayal of the data consistent. An ideal tool for changing the level of detail can be compared with an intelligent magnifying glass that determines whether an object matters or not and shows only as much detail as is relevant for a specific task (Frank and Timpf 1994). With Pad++ and Jazz, Benderson *et al*. (1996) introduced dynamic multi-scale user interfaces that allow people to freely pan and zoom. When an object is selected, its level of representation is raised and simultaneously the level of other objects is lowered. This change is reflected by the size of an object and by the amount of information that is visible. Since the change of detail is an action that is frequently needed, it must be executed automatically by the system. A manual change might be necessary to adjust the level of detail, for instance, if a representation is too complex.

Guideline 11: Pointing and talking are suitable modalities to change views of information due to their simplicity and efficiency. The use of drawing gestures is appropriate for zooming into parts of a user interface.

3.2.5 Operating

To operate a computer system means to have control over hardware and software, being able to coordinate the different processes, and to maintain the system in a stable state. The system provides users and applications with a basic framework of functions and communication interfaces. From the user's point of view, the interaction necessary to operate a system must be reduced so that the level of distraction is low and the focus is on the actual task to perform. Operating is limited to the following three actions: start, stop, and switch (Figure 3.1).

Start, stop, and switch focus on applications and processes. Due to the simplicity of these actions, the appropriate modality is pointing, eventually supported by verbal interaction, which is very similar to changing views of the previous section. The use of writing should be limited to initial configuration tasks; however, more complex modalities should not be used once the system operates under normal conditions.

Guideline 12: *Pointing, supported by simple verbal statements, is the modality of choice for operating computer systems. Writing or typing can be used for configuration purposes.*

3.2.6 User Actions in a Multimodal User Interface

Table 3.1 summarizes the guidelines for the suitability of individual user modalities for specific user actions. The expressiveness and flexibility of a user interaction can be improved if drawing and talking are included into the set of possible user modalities. Compose actions will benefit directly from these alternative modalities, while actions focusing on perception, selection, and operation are provided with a secondary modality that can be used as a backup if the application of the primary modality is impossible (e.g., because of a disability of the user).

User Modalities		**User Actions**							
		Perceive	**Select**	**Compose Text**		**Compose Drawings**		**Change View**	**Operate**
				Simple	Complex	Simple	Complex		
Output:	Pointing		**High**	Low	None	Medium	Low	**High**	**High**
	Writing		Low	**High**	**High**	Low	None	Low	Low
	Drawing		Medium	Low	None	**High**	**High**	None	None
	Talking		Medium	**High**	Medium	Low	None	Medium	Medium
Input:	Seeing	**High**							
	Hearing	Medium							

Table 3.1 Suitability of individual user modalities for typical user actions.

3.3 Summary

This chapter explored the potential of various *user modalities* and suggested guidelines for their applicability to individual user actions. The set of elementary *user actions* includes *perceiving*, *selecting*, *composing*, *changing views*, and *operating*. More complex *user operations* are built on these elementary user actions. Sketching is a valuable modality for the generalization, interpretation, and visualization of spatial scenarios or configurations. A direct interaction is a key issue in a sketch-based user interface. The use of modalities, such as drawing and talking, leads to easy-to-use systems that are powerful and effective to operate. Considerable advantages can be expected for three database-related user operations in a GIS, including *browsing*, *querying*, and *updating* (Blaser 1997; Egenhofer and Kuhn 1999). While browsing and updating operations are primarily simplified by using sketches and gestures, querying a spatial database experiences a boost of expressiveness. Hence, spatial queries that are otherwise difficult to formulate can be reduced to a simple sketch and the translation and interpretation of the user's mental model of the query into a processable form is delegated to the system.

Chapter 4

A Survey of People’s Sketching Habits

Although people’s sketching habits differ considerably from one person to another, most everybody is able to draw a sketch or understand a sketched scene that was drawn by somebody else. The reason for this common base of interpretation may be a set of reoccurring structures, pattern, symbols, or sketching strategies that people use when they draw a sketch. It can also be expected that sketches that are generated within the same application domain have a strong inter correlation.

The primary motivation for studying the sketching behavior of people is that a thorough understanding of people’s sketching habits is an essential requirement for the development of techniques that allow an automated interpretation of freehand sketches. The focus of this investigation is, therefore, on collecting and analyzing information about how people sketch. The goal is to acquire a solid base knowledge for the development of sketch-based applications.

For the purpose of our analysis we have conducted a survey, in which subjects were asked to draw freehand sketches according to different written scenarios. All sketches have been reviewed and analyzed (Blaser 1998). This chapter describes the setup of the survey, it provides a synopsis of our observations, and interprets the results.

4.1 Setup

Each surveyed subject was asked to draw three sketches based on a written task description–one for each sketch. Subjects were also asked some task-related questions for each sketch and some general questions at the end. On average it took the subjects approximately one hour to complete the survey. The survey was sent by mail and it came with a complete set of instructions. Of the 56 individuals that were asked to participate, 32 subjects completed the survey. Five additional surveys were sent out prior to the actual survey to assess the quality of the survey and to get a first impression of the results that could be expected. These five surveys were not included in the final interpretation of the sketching survey.

4.1.1 Surveyed Subjects

The selected group of subjects can be divided into those that are familiar with GISs and those that are not. The primary group consisted of students and faculty of the Department of Spatial Information Science and Engineering at Orono. The second group included individuals from the US, Germany, Switzerland, and India with various professional and cultural backgrounds. The groups included 11 female and 21 male subjects. The age of the surveyed individuals ranged from 25 to 57 years. Of the possible total of 96 sketches 91 sketches were evaluated and interpreted. The five remaining sketches were either not drawn or inadequate for an interpretation.

4.1.2 The Survey

The survey was prepared in English and German with identical content. This was necessary, because of the international setup so that every subject could read the survey in his or her mother language. The survey included the following elements:

- A cover letter with a general description and an explanation of the purpose of the survey.
- A page with general instructions that explained how to complete the survey.

- A *written description* for each of the three *sketching problems,* each with six related questions as well as some space for further comments.
- A *printed screen* for each sketching problem that could be used to draw the sketch. This page was a screenshot of a mockup of a sketch-based application.
- One page with *general questions*.

Each of the three sketching tasks was setup the same way: Next to the problem description and the sketching area was an empty list where the subjects were asked to specify the chronological sequence in which they drew their objects. This list and the six questions had to be completed after the sketch was drawn. The three sketching scenarios have different purposes and contexts.

The first scenario (*Familiar Scenario*) was about sketching a situation with which the subject is well acquainted. It was assumed that the subject would have more knowledge about the spatial situation than he or she would actually draw in a sketch (Blades 1990). Hence, one could assume that such a sketch was likely to contain only a set of selected and prominent objects that are essential for the description of the scene. One could further anticipate that this would lead to a meaningful and consistent representation of reality.

The second scenario (*Unfamiliar Scenario*) asked the subjects to draw a sketch of a spatial situation with which they were unfamiliar. Here we expected to obtain sketches that are fragmented and unreliable. Because of the subject's unfamiliarity with the environment, objects and landmarks are likely to differ considerably from those in the *Familiar Scenario* (Lynch 1960). Spatial or representational errors are also more likely to occur within such a setup.

The final sketching problem (*Imaginary Scenario*) requested the subjects to create an imaginary spatial scene, solely based on a written description. The description was intentionally ambiguous and some important information was missing. Sketching this scenario was further complicated, because the description included some unusual objects, such as topographic elements (e.g., mountain) and time as an expression of a distance. The interpretation and drawing of such a sketch required a great deal of imagination and it provided insights about how people transform a written scene description into a graphical sketch.

4.1.3 Analytical Setup

The evaluation of the sketches focuses on elementary structures, such as sketched objects, binary relations, and object annotations, to reduce the initial complexity of the analysis of our survey. Hence, the three sketches of each subject are analyzed one after another and object-by-object. The assessment of each sketch is done manually and the intermediate results are stored in a Microsoft Access database consisting of four tables with forms to simplify the input.

The main table contains information about each detected object (44 classification parameters for each object). Classification schemas with predefined interpretation guidelines for the recorded parameters and transparent rulers to assess metric and directional information are used to obtain consistent results during the evaluation of the sketches. Each object can have multiple written annotations, which are stored in a separate table. All objects in a sketch are connected through a table that stores general information about each sketch. The three sketches of each person are linked to each other with the forth table storing the subject's answers to the questionnaire.

The entire set of surveys is examined by the same person so that the interpretation is consistent. The analysis and interpretation of the intermediate results in the database are conducted using SQL to query the database (MS Access), and MS Excel and MathCAD to visualize the results.

4.2 Ingredients of a Sketch

The analysis focuses first on elementary building blocks of sketches. Three basic elements of a sketch can be distinguished. On an abstract level a sketch is a collection of strokes. These strokes, however, are typically not considered individually, but grouped together and perceived as *sketched objects* that have specific spatial or conceptual relationships to each other. Our approach is, therefore, to investigate objects and their *relations* within sketches. Based on the geometry of sketched objects and their spatial configuration it is possible to describe the topological, metrical, and directional concepts of a sketch. This information is also sufficient to compare different spatial scenarios with each other and to assess their similarity.

The expressiveness of objects and relations can be enhanced if their semantics is taken into account as well. Hence, in order to add a specific meaning to an object or to avoid drawing complex objects, people frequently use written or spoken annotations. Accordingly our analysis focuses on three basic components: *sketched objects*, *relations*, and *annotations*. The following three sub-sections describe the results of our evaluation of these building blocks of a sketch.

4.2.1 Sketched Objects

Sketched objects are the logical entities in a sketch. Individual sketched objects are the result of a meaningful interpretation of the set of sketched strokes. Objects can be composed of multiple intersecting or non-intersecting strokes. It is also possible that an object contains no drawn elements, for instance, if it is an annotation. Such objects can be defined as *virtual objects*. Because objects are logical entities, an object may enclose multiple independent *components*. In analogy to the object-oriented approach, objects that are hosting other objects are referred to as *composite-objects*. An example of a composite-object is a town containing houses, where houses are the components.

Drawn objects are the primary building blocks in a sketch. Hence, by knowing only the type of objects that populate a sketch it may already be possible to reveal the sketch's meaning. From this point of view information about objects can also be seen as metadata of a sketch. The term *sketched object* stands for a multitude of differently drawn and non-drawn real-world representations, as there are no strict rules of how to represent real-world objects in a sketch. For instance, someone's home could be described with a perceptive drawing of a house, a circle, a square, a front view, or one could just write *my house* without drawing anything. It is not trivial to perceive the meaning of sketched objects without some basic knowledge about how people sketch. On the other hand, we assert that there exist some common patterns of how people sketch, because otherwise nobody could interpret other people's sketches.

4.2.2 Object Relations

A relation is the virtual link between two or more objects; a binary relation links exactly two objects. Higher relations can occur when, for instance, a poplar tree in an avenue is standing *in line* with its neighboring trees. A hierarchical relationship involves a set of objects that can either be abstracted into a higher-level object or that have a specific relation to another higher-level object. In both cases it is possible to break such a situation down into multiple binary relations between components and the composite objects. Another possibility is to use grouping mechanisms to describe relations between objects standing in a hierarchical relationship with each other. For our investigations we consider primarily *spatial relations*. While the number of possible binary relations between n objects grows exponentially (Equation 4.1), one can assume that only a subset of all binary relations is necessary and relevant for a robust interpretation of a sketch.

$$\text{Number of binary Relations:} \quad \frac{n^2 - n}{2} \tag{4.1}$$

For instance, there is most likely no direct relationship between two objects that are spatially disjoint, drawn on opposite sides of a sketch, and that have multiple other objects in between. For our analysis we take, therefore, only *spatial neighborhood relations* into account (Tobler 1970), because they capture fundamental geographic concepts and are more likely to be essential for the intended message of a sketch.

4.2.3 Annotations

An annotation is a written or spoken verbal note that describes an object, a group of objects, or a relation between two objects. If an annotation is used without referring to an object, it can be seen as a *virtual object*. Annotations describe characteristics of an object that cannot be formulated graphically, such as an address or a name of a building. Annotations can also be used to define specific properties of a relation between two objects. Specifying the time or distance to get from A to B is an example. Typically annotations and sketched objects are used in a complementary sense; however, they can also contradict each other (Egenhofer 1996a).

Every sketch can be split into objects, binary spatial relations, and annotations. However, annotations are optional, while objects are mandatory and spatial relations are implied when more than one object is drawn.

4.3 Analysis of Sketched Objects

This initial evaluation focuses on sketched objects with respect to their *class*, *portrayal*, and *purpose*. The classification is conducted using an assessment schema that was *a priori* defined.

4.3.1 Object Classes

Since all sketches of the survey belonged to the same geo-spatial domain it was possible to define a set of object classes that covers most cases. An *object class* is defined as a category of objects with similar characteristics, such as the class of *building* objects (Rodríguez *et al.* 1999). Each object class may have multiple subclasses with a more specific description. The *School* class, for instance, is a specific subclass of the building class.

The total number of sketched objects analyzed is 832 (69% of all objects analyzed in the survey) and their distribution with respect to the 19 object classes is depicted in Figure 4.1. This classification includes all objects of the *Familiar* and *Unfamiliar*

Scenario. The evaluation excludes all objects of the *Imaginary Scenario*, because this scenario included an explicit list of objects that had to be drawn. The two other sketching problems are less biased by the problem description.

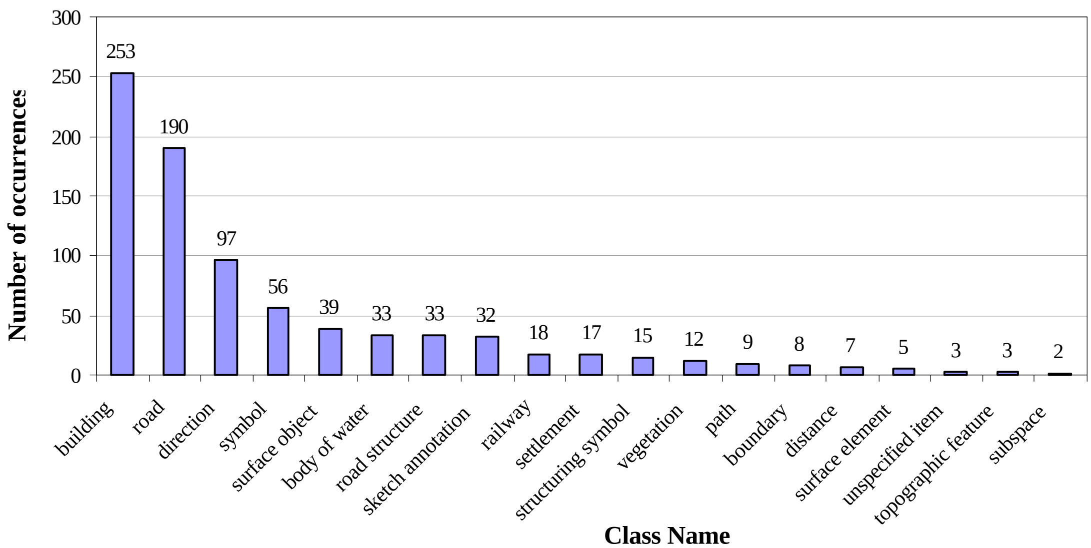


Figure 4.1 Frequency distribution of the 19 object classes, sorted by size (*Familiar* and *Unfamiliar Scenario*).

- The two most frequently used classes (*building* and *road*) cover 53% of all sketched objects; the first nine most frequent classes make up 90% of all objects. Depending on the field of application and the context of the sketch these classes will, of course, vary; however, we can expect that for a specific field of application there is only a limited number of object classes necessary to interpret a sketched scene. Besides such domain specific sets of expressions, there exists also a generic sketching terminology, which is domain independent. This standard set of object classes includes generic symbols, such as arrows or connecting lines (*direction*, *symbol*, and *distance* object class in Figure 4.1).

- Some object classes, such as *buildings* have as many as eleven subclasses, while other important object classes, do not show such a great diversity. The *road* class (*streets*) and the *body of water* class (*rivers*) are two examples.

- Natural objects, such as *body of water* (4%) or *vegetation* (1%), make up only a small part of the entire set of objects found in the *Familiar* and *Unfamiliar Scenario*. Artificial objects, such as *buildings* (30%) or *roads* (23%), on the other hand, occur more frequently.
- Another interesting observation can be made with respect to objects conveying metric information. Only eight objects or less than 1% of all objects in the *Familiar* and *Unfamiliar Scenario* carry explicit metric information. Hence, most sketches use primarily topology and the arrangement of objects to describe a specific situation. This finding supports the earlier observation concerning the relations between objects in geographic space that *topology matters and metric refines* (Egenhofer and Mark 1995).
- A similar observation can be made about topographic features in sketches. Only one person in the survey used topographic structures, such as a hill or a valley. Hence, it appears that people try to keep their sketches *flat*, like a map (Willauer 1993). There are two possible explanations for this observation. First, people may lack an appropriate representation for topographic features and second, they may simply not need topographic objects for an adequate description of a spatial scene–similarly to other natural objects.

4.3.2 Portrayal

Sketched objects are typically abstract and generalized representations of their real-world counterparts. Hence, a typical sketched object consists only of few line strokes that are frequently approximations and single-colored. This section analyzes different methods of object portrayals within sketches and attempts to develop an overview of how people visually express objects in spatial sketches.

4.3.2.1 Shape

The shape of sketched objects is evaluated from two different points of view: first with respect to realism and second with respect to the type of abstraction. According to this classification, an object can fall into either of the following shape classes: *symbolic, semi-symbolic*, and *realistic.*

Objects are considered *symbolic* if their representation is based on a symbolic representation of an object that has nothing in common with the actual look of the object in reality (Figures 4.2a and 4.2d). The association between sketched object and the original object is made through a symbol and not through the shape of the sketched object. *Realistic* objects, conversely, try to capture reality with the expression of unique or distinguishable features of an object (Figures 4.2c and 4.2f). Finally, objects that have both symbolic and realistic characteristics fall into the *semi-symbolic* category (Figures 4.2b and 4.2e).

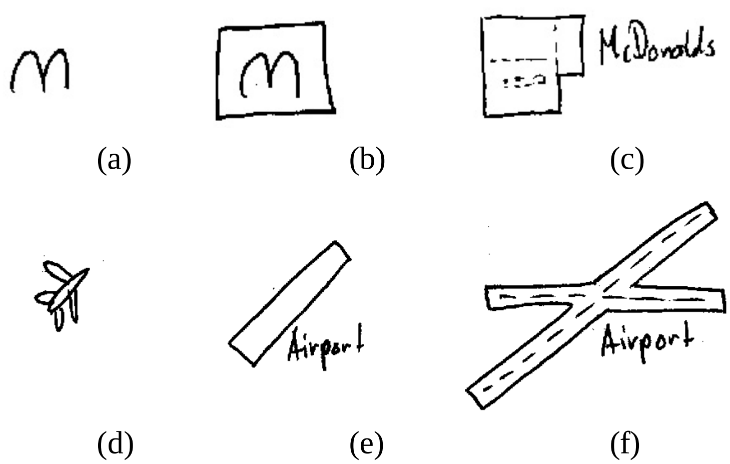


Figure 4.2 Two sequences of sketched objects with the same semantics but different portrayals.

Most objects were classified as either symbolic (42%) or semi-realistic (56%), with only few objects of type realistic (2%). This distribution does not change significantly if objects from the *Imaginary Scenario* are included.

Considering the type of abstraction of sketched objects one can further differentiate between line and region objects. Objects with primarily line characteristics are classified as *straight, curved,* or *complex*. Region objects can be of type *square, box, circle, oval,* *cross*, or *complex*. It is possible for one object to have components from multiple shape classes. Figure 4.3 reveals details about the distribution of object shapes.

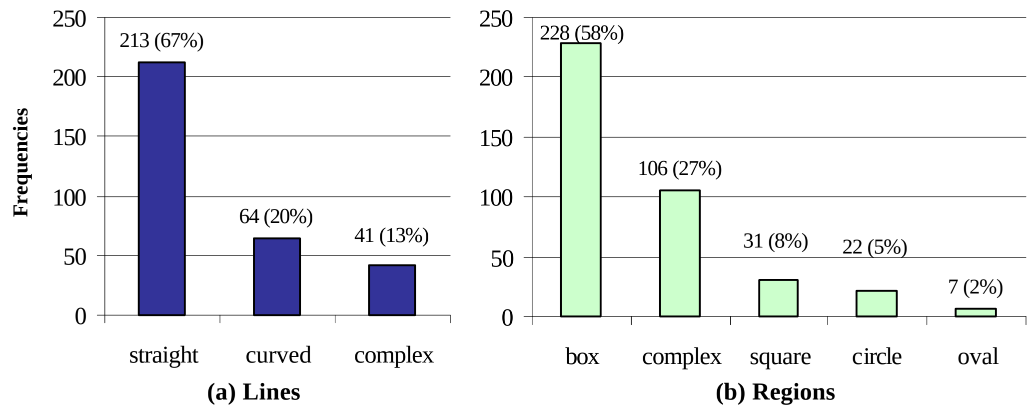


Figure 4.3 Frequency graph of the shapes of (a) lines and (b) regions (*Familiar* and *Unfamiliar Scenario*).

More than half of all objects of the *Familiar* and *Unfamiliar Scenario* are represented by simple shape forms, such as straight lines or boxes (62% of all objects with a classifiable shape). Squares, circles, and ovals are used less frequently. Excluding shape categories that contain complex structures reduces the set of possible object shapes to line, curved line, box, square, circle, and oval, which cover 78% of all sketched objects in the *Familiar* and *Unfamiliar Scenario*. This observation is important, because it is another indication that people tend to keep their sketches simple and their objects abstract. This finding also indicates that the context and the actual configuration of a sketch are more important than the representation of single objects (i.e., sketched objects taken out of their context have frequently no own meaning).

4.3.2.2 Type of Outline

The outline of a sketched object indicates something about a person's sketching technique. There were six distinct outline types in the surveyed sketches (Figure 4.4). These outlines can by used to classify line and region objects.

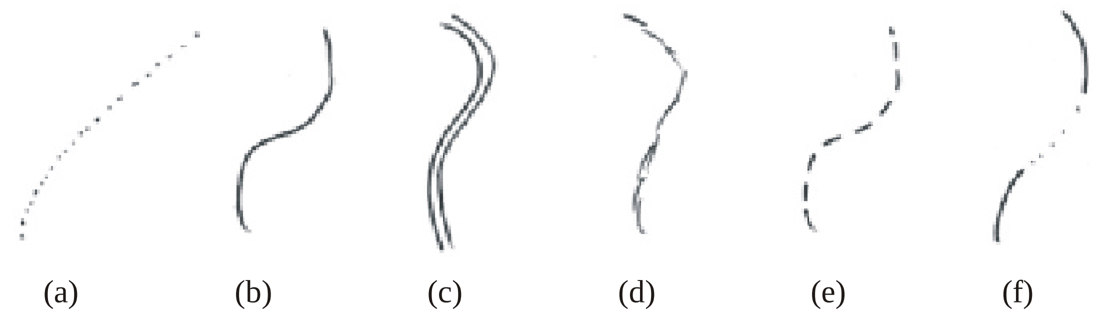


Figure 4.4 Six different outline styles: (a) doted, (b) simple line, (c) double-line, (d) multi-stroke line, (e) dashed, and (f) mixed.

Figure 4.5 shows the frequencies of the six different outline types over all three scenarios. Simple lines are most frequent (79%), followed by double-lines (9%). Simple lines are used for objects throughout the entire spectrum of object classes, whereas other line types are correlated with specific object types. Double-lines, for instance, are frequently used for waterways, but were never used for boundaries. Multi-stroke lines seem to indicate an individual's drawing style.

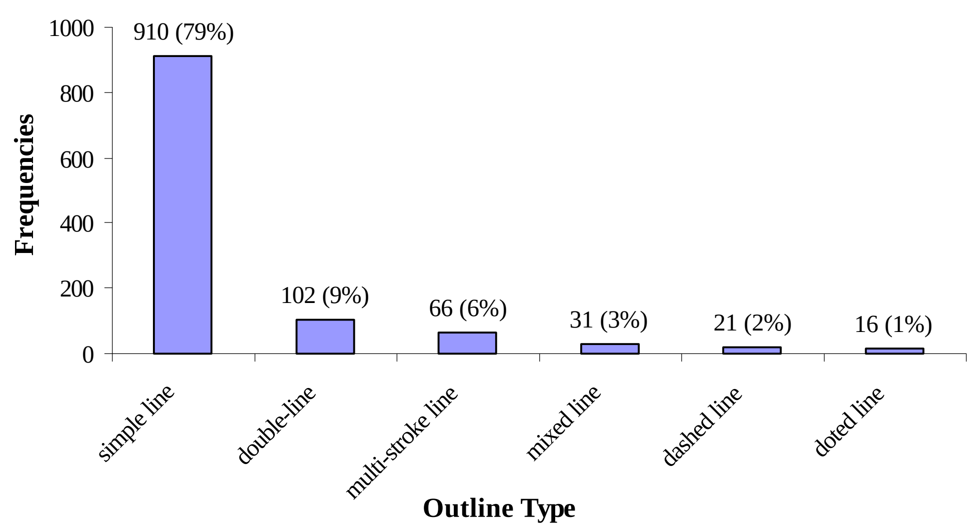


Figure 4.5 Histogram with the occurrence frequencies of the six outline-types (All scenarios).

Despite the predominance of simple lines there are other outline types that are important for certain object classes. For instance, bodies of water, such as rivers or brooks, use double-lines in 37% of all cases. Double-lines are also often used for roads (24%) or railways (28%). Other object classes, such as paths (31%) and boundaries (22%), are sometimes drawn as dashed lines. These observations indicate that there is a relationship between an object's type and its outline.

4.3.2.3 Number of Strokes per Sketched Objects

The number of strokes per sketched object is constant and independent from the sketching scenario according to our observations and Figure 4.6. Most sketched objects have only two strokes. The average number of strokes per object is between five and eight.

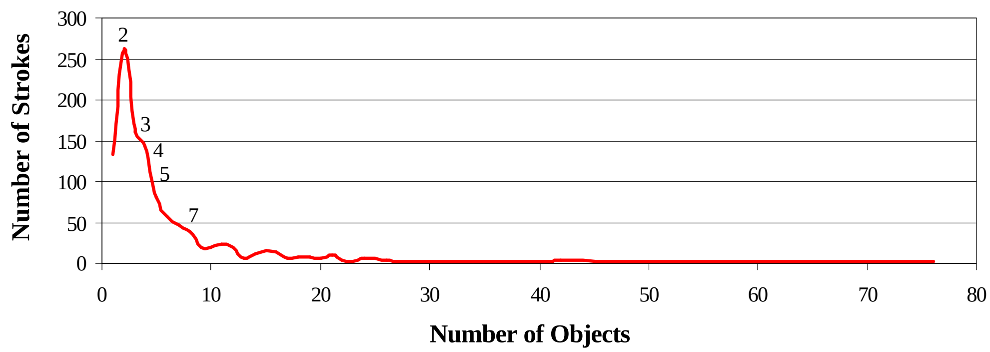


Figure 4.6 Number of strokes per object (All scenarios).

The number of strokes per object increases if the sketch complexity increases or if unusual objects are drawn, such as in the *Imaginary Scenario* (Blaser 1998). Conversely, if a sketch is simple or if objects are vague then the average number of strokes per object tends to decrease.

4.3.2.4 Completeness of Objects

Considering the completeness of objects, one can distinguish between *complete*, *partial complete*, and *incomplete* objects. The survey revealed that people tend to sketch carefully. (Table 4.1). However, some individuals sketched consistently incomplete objects–an observation that may be attributed to their sketching technique.

Count	%	Category
538	50%	complete
487	45%	partial complete
53	5%	incomplete

Table 4.1 Average distribution of objects with respect to their completeness (All scenarios).

With respect to the individual sketching scenarios, people seem to sketch more frequently incomplete objects if they are foreign with an environment, such as in the *Unfamiliar Scenario*.

4.3.2.5 Perspective

In general people stay within the second dimension when they draw sketches (Willauer 1993). Under certain circumstances, however, the surveyed subjects used perspective representations of objects. There are three distinguishable types of perspectives in the surveyed sketches: *front elevation*, *perspective 3D*, and *mixed representation*. In the *Familiar* and *Unfamiliar Scenario* people made only moderate use of perspectives (5% of all objects). However, the use of perspective objects increases considerably in the *Imaginary Scenario* (23%). Examples of objects that are frequently drawn in perspective are cars, trains, traffic lights, or topographic structures, such as mountains. Of the three perspective types distinguished in our survey, the front elevation is the most frequently used, followed by the 3D perspective. Mixed perspectives are rarely used.

4.3.2.6 Virtual Objects

A virtual object is an object that has no drawn elements; instead, it is defined by a written annotation. All sketches of the survey contained a total of 54 virtual objects (approximately 4.5% of all sketched objects). Twenty-four individuals (77%) used at least one virtual object within their sketches. Virtual objects are distributed over 36 of a total of 91 sketches (40%). There is no sketch with more than three virtual objects. The average is 1.5 virtual objects per sketch if virtual objects are used at all (with $\sigma = 0.6$). These numbers show that virtual objects are frequently used, but that their density per sketch is low. Virtual objects are often used to specify extended, area-like objects that are composed of different components or that are otherwise difficult to describe. Virtual objects can also be used to orient sketches, for instance, when a remote place is used to establish a reference direction for the sketch.

4.4 Spatial Relations

The term *relation* can be defined as "*Natural, logical, or virtual association between two or more things, that are relevant to one another* (Microsoft 1999)." The interpretation of a relation is generally based on people's perception of a situation in reality, which makes a relation subjective, because reality itself depends on perception *and* interpretation. To be able to describe relations between things more objectively, theories have been developed that focus on specific characteristics of a relation. Theories about *spatial relations* aim to formally describe the relation between objects on a geometrical basis. A *binary spatial relation* is the special case, where only two objects are involved. Because of their simplicity and their elementary character, binary relations are often the preferred type of spatial relations.

4.4.1 Topology

The evaluation of topology is based on the 9-intersection (Egenhofer and Al-Taha 1992). In this context all sketched objects are considered as regions so that there are eight possible topological relationships. In the case of ambiguous situations, the subject's intention is considered and the intended relation is translated into the appropriate topological term. According to this interpretation, the road in Figure 4.7 meets with four houses. House (a) is considered disjoint, but it still carries the attributes along and parallel.

Figure 4.7 Sketch with five buildings along a road; four of them qualify for a *meet* condition (b) - (e), one house is disjoint (a).

Each object has n-1 binary topological relations with other objects. The total number of possible relations in a sketch can be computed by using Equation 4.1. Non-disjoint relations are relevant, because they indicate a *physical* connection between objects (Florence 1997). Disjoint relations are more difficult to classify (Shariff 1996). The evaluation of the topology focuses, therefore, on all non-disjoint relation between sketch object pairs (Figure 4.8).

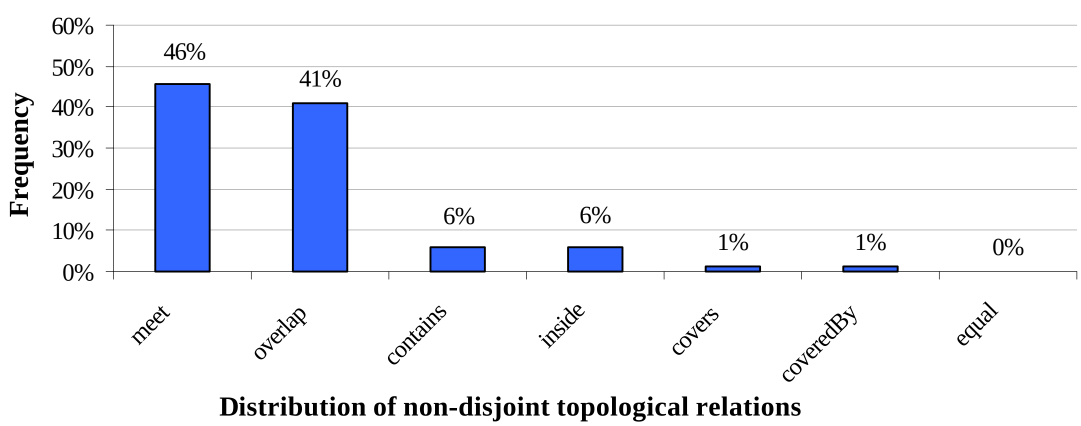


Figure 4.8 Frequency analysis of the recorded binary topological relations (*Familiar* and *Unfamiliar Scenario*).

Approximately two thirds of all objects (62%) stand in at least one non-disjoint relation with another object. This indicates that spatial sketches are interlinked structures that form topologically connected networks. The non-disjoint relations that were recorded during the analysis of the sketches represent 8.2% of the total possible number of binary relations. Figure 4.8 shows their distribution. The majority of spatial relations in the survey are of type *meet* and *overlap*. These relations are typically used to connect objects. Relations that represent containment (contains, inside, covers, coveredBy) were rarely used.

4.4.2 Spatial Location

The spatial location of objects was recorded by using a regular 4x4 grid, in which an object was associated with one or more of 16 rectangular areas. Figure 4.9 shows the distribution of the spatial location of objects with respect to the sketching device. Most sketched objects are drawn in the center, with a continuous decrease in density towards the borders of the drawing area. The highest object density is found in the left center (Partitions 1 and 4).

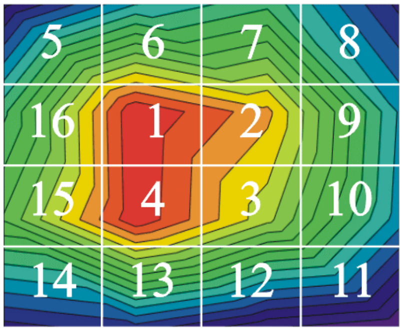


Figure 4.9 Contour plot reflecting the distribution of objects over all three scenarios. Red indicates the highest and dark blue the lowest density of objects.

4.4.3 Orientation

An object's orientation is its directional relation with a referencing system. The simplest frame of reference is the drawing device. Other referencing systems are the principal drawing direction of a sketch or that of a group of objects. The orientation of objects in our evaluation is measured with respect to the drawing device. The orientation values are manually assessed and range from 0° to 180° degree (with 10° increments). Of the 1208 objects analyzed, 993 objects (82%) have at least one prominent orientation. On average each object has 1.2 orientation indications. Figure 4.10 shows the orientation graph for all objects from all scenarios. The values from 180° to 360° corresponds to those between 0° to 180°.

Orientation of Objects	Count	%
0°	**312**	**26%**
10°	40	3%
20°	40	3%
30°	27	2%
40°	16	1%
50°	24	2%
60°	20	2%
70°	35	3%
80°	42	3%
90°	**438**	**36%**
100°	41	3%
110°	22	2%
120°	32	3%
130°	16	1%
140°	19	2%
150°	20	2%
160°	16	1%
170°	26	2%

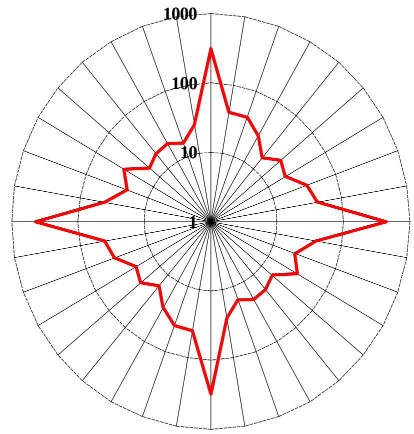


Figure 4.10 Object alignment with respect to the drawing device (All scenarios).

An alignment towards the North-South and East-West axis are the prevailing orientations. About 40% more objects are drawn with an East-West than with a North-South orientation. The distribution between the two main axes is regular, although there are slightly more objects with an orientation between 10° and 80° compared to those with an orientation between 100° and 170°.

4.4.4 Direction

In this section we examine objects that have one or more pointing directions (i.e., a direction that is explicitly indicated with an arrow type symbol). A direction can be implicit or explicit and an object can have more than one direction. In total there are 323 (27%) objects that qualified for this analysis. This number does also include objects that have a deduced direction, for instance (i.e., objects that inherited a direction from an associated arrow). Figure 4.11 depicts some possible configurations of objects with an indicated direction. Figure 4.11d is particularly interesting, because it is ambiguous. The place named *Boston* could be at the end of the arrow, but it may as well lie outside the drawing area in the indicated direction.

(a) (b) (c) (d)

Figure 4.11 Objects with an indicated direction: (a) a north arrow, (b) a street with an explicit flow or path direction, (c) a view symbol indicating the direction of the view, and (d) a virtual object with an indicated direction.

The referencing system for the assessment of directions is the drawing device. Figure 4.12 shows the spectrum of possible object directions and their frequencies. To reflect the different types of objects with an indicated direction the graph distinguishes between (1) all direction objects, (2) all direction objects with a North direction, and (3) all objects with a direction (i.e., except those with a North direction). The distribution of indicated directions is similar to the orientation of objects in that cardinal directions show a significant higher frequency than non-cardinal directions.

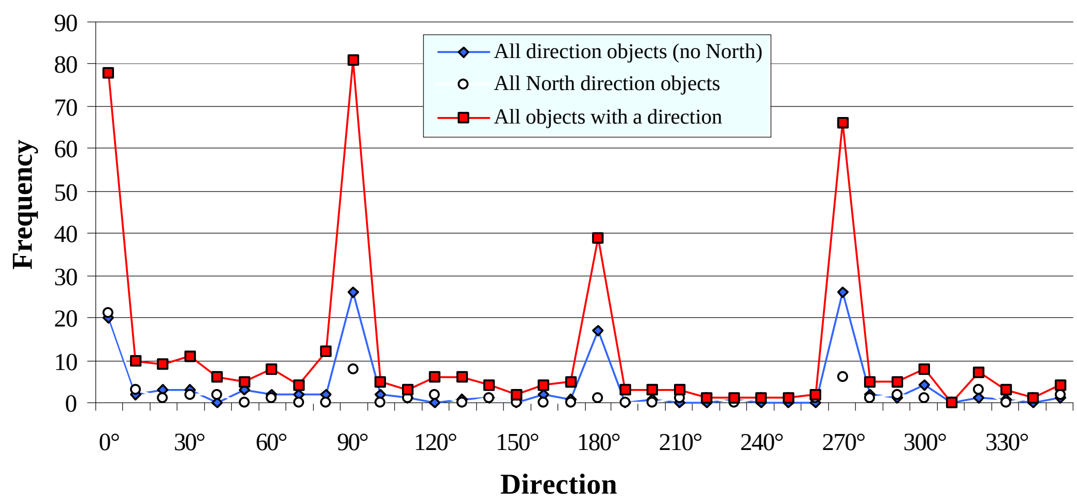


Figure 4.12 Distribution of indicated directions for all objects (All scenarios).

Figure 4.12 shows that objects pointing to the North and to the East are twice as frequently used as objects pointing to the South. Those objects with a westerly direction score somewhere in between. The preferred non-cardinal directions lie between 0° and 90° (16%), while the three remaining sectors score only between 4% and 8% of the total number of directions. This observation supports the assumptions that many people have an inclination to write or sketch objects slightly tilted upwards, from left to right.

Focusing on North directions it becomes evident that most of the subjects orient their sketches by providing a north direction pointing towards the top of the drawing device (33%). The distribution of not-North directions is more symmetric compared to that of objects pointing North. The number of objects pointing East and West are similar, whereas objects with North and South directions are slightly less frequent.

4.4.5 Parallelity and Orthogonality

Human-built objects, such as buildings, roads, or malls that are built closely to each other are frequently in a specific angular configuration. To verify this observation we counted for each object the number of adjacent objects that were either parallel or orthogonal to the object in question. In order to qualify as an adjacent object, an object has to be in direct sight and its distance to the referring object may not exceed this object's maximal dimension. Figure 4.13 indicates that sketched objects are frequently parallel or orthogonal to each other (particular for the *Familiar* and *Unfamiliar Scenario*). Two third of all objects had one or more neighboring objects that were either parallel or orthogonal. The *Imaginary Scenario* had significantly less such angular conditions, but still, one out of three objects had at least one parallel or orthogonal neighbor.

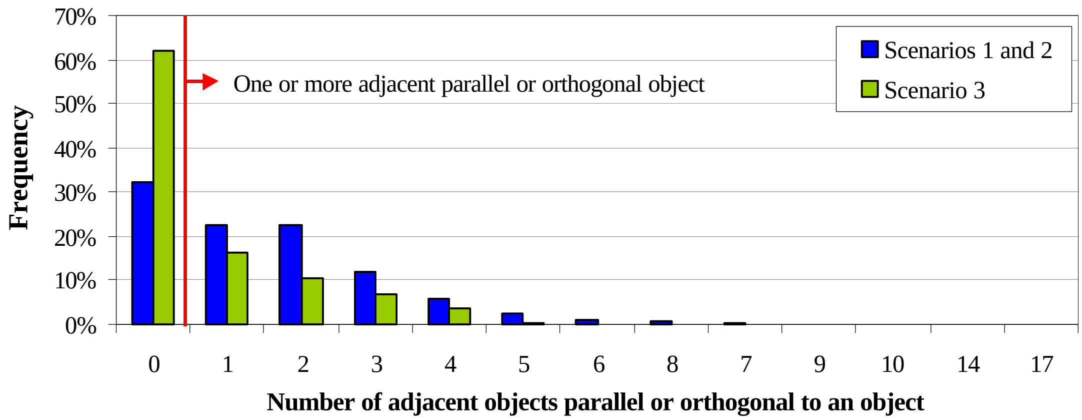


Figure 4.13 Frequency of objects with parallel or orthogonal neighbors (All scenarios).

The implication of these observations is twofold. First, the arrangement of objects in a parallel or orthogonal manner appears to be a frequently used concept in freehand sketches. In sketches that represent urban settings, such as the *Familiar* and *Unfamiliar Scenario*, this concept seems to be more dominant than in natural settings with less human influence.

Second, Figure 4.13 suggests that the majority of objects in a sketch are connected to their immediate neighborhood and that the connection between neighboring objects in this context is made either via topology or by applying certain metric conditions, such as closeness, parallelity, or orthogonality. The rational for this statement is that in order for an object to be considered in this evaluation it had to be close to the referring object. On the other hand we can assume that not every adjacent object is parallel or orthogonal to its neighbor. Thus, the number of objects that are close to each other is greater than or equal to the sum of all objects with at least one parallel or orthogonal neighbor. In addition to objects that are parallel or orthogonal to an object one may also count those objects to its neighborhood that meet or overlap this particular object. These considerations suggest that such *neighborhood relations* play an essential role within spatial sketches and that not all binary relations within a sketch are equally important.

4.5 Annotations

Written annotations are a frequently used form of assigning additional meaning to sketched objects. Approximately 60% of all objects in the survey have at least one written annotation. However, most people found that one annotation per object is enough (87%) and only few sketched objects have more than one written annotation. Object of certain purpose classes and of certain types are more frequently annotated than others. For instance, objects of the *distance* and *settlement* classes are typically annotated, while there is in general no annotation for symbolic objects or objects with a directional purpose. The analysis of the survey suggests that there are four primary reasons why people annotate objects in sketches:

- *Complexity*

  The meaning of an object with a complex semantics or an object that is difficult to circumscribe by graphical means alone is easier to convey when a written annotation is added.

- *Significance*

  Objects that are of superior importance within a sketch, such as start or end points, have often written annotations that can help to bring an object into focus.

- *Ambiguity*

  If a sketch contains multiple objects that share a similar appearance, then an annotation can be used to distinguish between such instances.

- *Simplicity*

  For some objects there is no adequate sketched representation; however, there may exist a commonly used term. A city's name is an example (e.g., New York).

Most written annotations in the survey are short, simple, and noun-based (70%) and only few individuals use entire sentences to describe a sketched scene (3%). The remaining annotations (27%) are short combinations of words, such as adjective or nouns. In regard of the content of written annotations, we found that the majority of annotations specified either name (36%), type (33%), or name and type (5%) of an object.

Most people place their annotations either outside (60%) or entirely inside of an object (33%). 50% of all annotations have the same orientation as their affiliated object, 30% are drawn in one of the cardinal directions of the drawing device. For the remaining annotations, there is no directional link to the affiliated object. The affiliation to an object is for the most part accomplished by a particular placement of the annotation in relation to an object. Only 17% of all annotations use linking symbols, such as arrows or connecting lines, for this purpose. People prefer to annotate their objects either immediately after drawing an object or later during the sketching process.

4.6 Summary

The following findings are relevant for an automatic assessment of spatial sketches.

- *People's sketches are simple and abstract.*

  People draw their environment (geographic space) with simple objects that have frequently no meaning if they are taken out of context. A typical sketch contains only a small number of objects. Sketched objects are highly abstract representations of their real-world counterpart. Simple lines and boxes are most frequently chosen for sketched objects. Objects that are of a particular significance, ambiguous, or for which there is no simple drawn representation are frequently annotated. People draw objects with clear boundaries and they prefer human-built over natural objects. Sketched objects were always used in a positive way, that is, there was no evidence that people use negation in their sketches. This is true for objects and their properties.

- *Topology matters while metric and orientation refine.*

  When people draw a sketch they are primarily concerned with topological issues between objects. In the same context people take also great caution when a specific object sequence or order is involved, such as when several objects are lined up along a road or when an object is between two other objects. The preferred topological non-disjoint relations in spatial sketches are non-containment relations, such as *meet* or *overlap*. For *disjoint* relations, topology alone is not expressive enough. In such cases people use metric and relative orientation to describe and refine object-object relations. Metric and directionality, in this context, are used in an implicit way. Explicit statements, such as verbal annotations, are rarely used.

- *Sketches are structured into object neighborhoods.*

  People tend to arrange and cluster objects such that they are connected. This connectedness is achieved by tying objects physically, through vicinity, special arrangements, or via context to one another. Physical links are expressed with *meet* and *overlap* conditions. Vicinity and special object arrangements are established by such concepts as parallelity, orthogonality, inline-ness, inbetween-ness, or similar forms of object arrangements. The context of a relation is more difficult to capture, because object-object relations must be evaluated within the overall context of the entire sketch. However, evaluating the general context of a sketch, based on the individual context of binary relations between object neighbors appears to be a viable approach for this purpose.

- *People have a specific sketching signature in their sketches.*

  Although we did not explicitly investigate individual sketching techniques of our subjects, we found similarities between the three sketches of each subject. This concerns the representation of specific objects types, the use of symbols, and their sketching style with respect, for instance, to detail or complexity. This observation can be seen as an indication that personalized profiles must be generated and used to automatically interpret and understand freehand sketches.

Chapter 5

The Digital Sketch

This chapter investigates the translation of the content of a sketch into a form that can be used to query a spatial database. The translation mechanism and the resulting *digital sketch* are key components of a sketch-based system. The goal of the translation is to capture the original intention of a user as accurately as possible. The remainder of this chapter describes the formal base for the prototype implementation of a sketch-based query system. After a review of formal models of binary spatial relations (Section 5.1), a symbolic representation for spatial sketches, the *digital sketch*, is introduced (Section 5.2). Section 5.3 discusses different approaches to improve the efficiency of the processing of the digital sketch. Section 5.4 presents a set of extensions of the formal model that enhance the expressiveness and stability of the digital sketch. The chapter closes with a detailed description of how digital sketches can be compared according to their similarity (Section 5.5). The result of this evaluation is the *scene similarity* between two sketched scenes, respectively between their digital sketches.

5.1 Formal Models of Spatial Relations

The digital sketch is based on formal models of space that were developed previously (Egenhofer 1996b). This section provides a brief summary of the formalisms used in this thesis.

5.1.1 *The 9-Intersection*

The 9-intersection is a qualitative formal model of space that describes the topology between points, lines, and areas. The model is based on point set topology, a theory that defines the rules between two point sets *A* and *B*, with *A* or *B* being either a point, line, or area. A point set of *A* has an interior ($A°$), a boundary (∂A), and an exterior (A^{-}). Topological relations between two point sets *A* and B are characterized by the intersection of *A*'s interior, boundary, and exterior with the interior, boundary, and exterior of *B* (Equation 5.1).

$$R(A,B) = \begin{pmatrix} A° \cap B° & A° \cap \partial B & A° \cap B^{-} \\ \partial A \cap B° & \partial A \cap \partial B & \partial A \cap B^{-} \\ A^{-} \cap B° & A^{-} \cap \partial B & A^{-} \cap B^{-} \end{pmatrix} \tag{5.1}$$

With each of these nine intersections being either empty or non-empty, the model has 512 possible topological relations between any two point-sets. However, some combinations cannot be realized (Egenhofer and Herring 1991). For two simple regions without holes that are embedded in R^2, this classification shows eight distinct topological relations (Egenhofer and Franzosa 1991) (Figure 5.1).

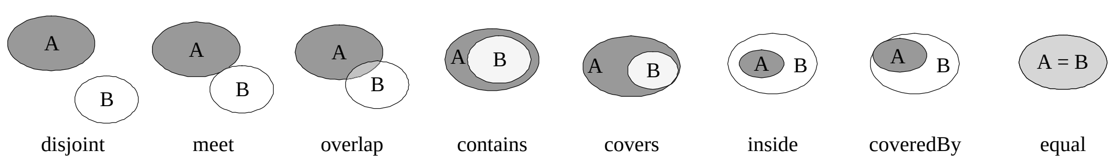


Figure 5.1 For two regions there are eight distinct topological relations.

Reasoning about gradual changes of topological relationships (Egenhofer and Al-Taha 1992) led to the formal model of *conceptual neighborhood of topological relations* (Figure 5.2).

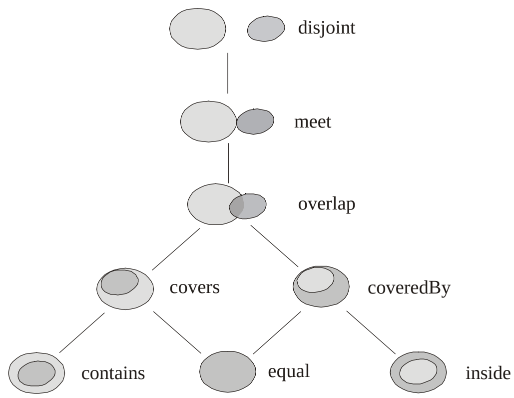


Figure 5.2 The conceptual neighborhood graph for the eight topological region-region relations.

The conceptual neighborhood graph describes the relationship between the binary topological relations of the 9-Intersection and orders them such that direct neighbors have the highest topological similarity. The model is, therefore, useful to assess the similarity of a pair of topological relations.

5.1.2 Metric Refinements of the 9-Intersection

While topology is important to perceive the global picture of a spatial scene, metric aspects become relevant if a scene is to be analyzed on a more detailed level (Egenhofer and Mark 1995). This is especially important for disjoint relations, because they are often insufficiently characterized by topology alone.

Shariff (1996) proposed several measures to capture the semantics of natural-language spatial relations between regions and lines through topological properties and metric refinements. These refinements include three quantitative distance concepts: *splitting*, *closeness*, and *fuzziness*. The number of possible refinements may vary with the geometric types of the objects involved (Egenhofer 1997a). Figure 5.3 shows eight possible refinements for region-region relations.

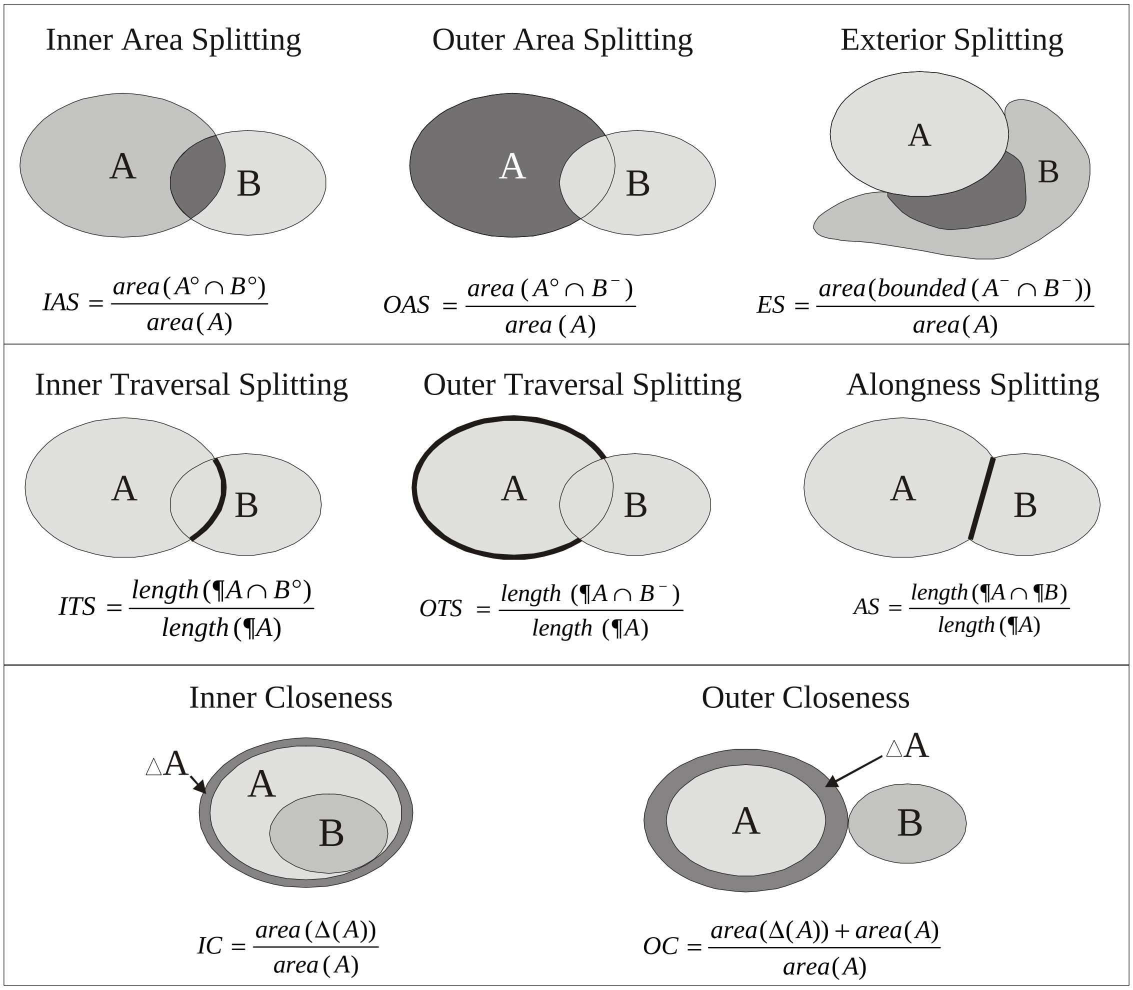


Figure 5.3 Metric refinements of topological relations between regions (Egenhofer 1997a).

5.1.3 Direction Relations using MBRs

A frequently used approximation of spatial objects is based on an object's projection onto the x- and y-axis of a coordinate system. In 2D space, this projection is a one-dimensional interval on one axis. If we consider that there are 13 possible relations between two one-dimensional temporal intervals (Allen 1983), then there are also 13 possible spatial relations between two objects for each dimension (i.e., considering only objects that are connected). Figure 5.4 shows the set of possible relations between two MBRs for the x-axis.

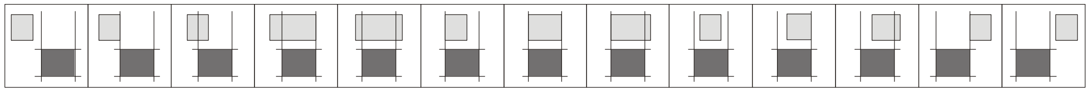


Figure 5.4 Possible relations between two MBRs along the x-axis.

The two-dimensional result is a 13x13 matrix that covers all possible cases (Papadias *et al.* 1995). Because every spatial object can be approximated by its MBR, this approach can easily be used to describe the direction relation between two objects.

5.2 Components of a Digital Sketch

The digital sketch is a meaningful representation of a sketch, reflecting important characteristics that are required to query a sketch against a spatial database. The model is based on findings in human-subject testing (Chapter 4) and on previous research (Section 5.1).

5.2.1 Model Components

The *digital sketch* consists of n distinguishable *sketched objects* (with $n \geq 1$) and m *binary spatial relations* (with $m = n \cdot (n-1)/2$) between them. Each sketch component has a specific set of attributes or properties (Figure 5.5).

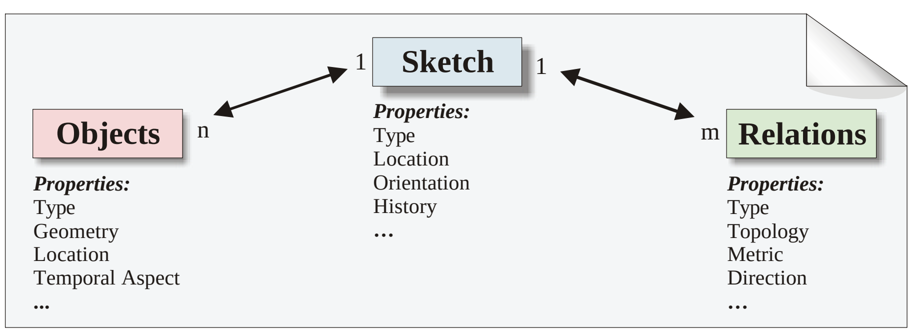
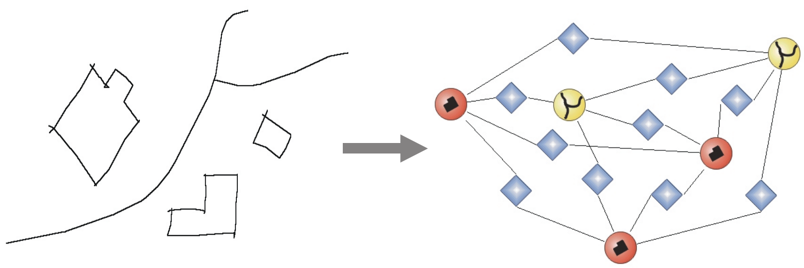


Figure 5.5 Relationship between the three basic components of a sketch.

The properties that characterize each component in a sketch consider qualitative and quantitative aspects. Sketched objects are described by geometric parameters. Spatial relations have a topological, directional, and metric component. The set of sketched objects represents the visual part of a sketched scene and the set of spatial relations between them describes a virtual network linking sketched objects with each other. Knowledge about both components is necessary to extract meaning from a sketch.

5.2.2 *Association Graph*

The virtual network between sketched objects can be described by an *association graph*. Each binary spatial relation is represented by a link or edge of the graph, and each sketched object by a node. Figure 5.6a depicts a sketched spatial scene and Figure 5.6b shows the corresponding diagrammatic representation (i.e., association graph). Sketched objects are shown as yellow and red circles, while binary spatial relations are represented as blue diamonds.



(a) (b)

Figure 5.6 (a) The sketch is translated into (b) a diagrammatic representation that interconnects all sketched objects.

The association graph in Figure 5.6b is complete, that is, each object is linked through a spatial relation with every other object. For every additional sketched object, `n-1` binary spatial relations have to be added to the graph (Equation 4.1), letting the association graph grow by $O(n^2)$.

5.2.3 *Discussion*

The complete association graph, involving all binary spatial relations, is stable, because every change in the configuration of the graph, such as the movement of an object, is propagated throughout the entire network to every node. This stability, however, comes at the price of a exponential increase of the number of binary spatial relations if sketched objects are added. This observation is relevant, because the effort to process a sketch, consisting of objects and binary spatial relations, increases by $O(n^2)$ as well. Hence, the complete association graph has an undesired characteristic of growing and it is, therefore, inappropriate to efficiently represent binary spatial relations within a sketch.

Another argument against the use of the complete approach is common sense. It seems unlikely that people take into account all possible spatial relations between objects when they interpret a sketched scene. We assert that this is done selectively, for instance, by considering context (e.g. semantics and hierarchies) or spatial neighborhood (i.e., spatial closeness) (Tobler 1970). With such an approach, it is possible to evaluate spatial relations between objects according to their relevance to the scene. An intelligent selection of a subset of binary spatial relations has many advantages over a comprehensive approach:

- Storing a digital sketch with a reduced association graph results in a smaller dataset that needs *less space* in the database of sketches, is faster to access, and *more efficient*.
- Processing digital sketches using a reduced association graphs (e.g., comparing two association graphs according to their similarity) will be *faster*, because the system has to take into account *fewer components*.
- The complete association graph is a highly over-determined system (Similar to a surveying network, where all possible angular and distance measurements have been assessed). Considering only a relevant subset of all binary relations leads to a *less redundant representation*.
- An association graph that considers only a subset of all binary relations is *easier to update*, because the system can restrict the update to those model components that are affected (i.e., linked through spatial relations to the object that is changed).
- An intelligently reduced association graph allows a *higher level of access*, because objects are already clustered in a meaningful way. For instance, if the mechanism to

evaluate the relevance of spatial relations is based on closeness, then Tobler's First Law of Geography (Tobler 1970) can be assumed ("everything is related to everything else, but near things are more related than distant things").

The development of methods that simplify the complexity of a digital sketch's association graph, while maintaining its expressiveness, is an important contribution towards the goal of enhancing the efficiency of the proposed symbolic model. The next section investigates different approaches to reach this goal.

5.3 Improving the Efficiency of the Digital Sketch

The efficiency of a digital sketch is directly linked to the complexity of the association graph. The fewer components the association graph has, the more efficient is the processing of the digital sketch. The model for a sketched scene consists of two components (i.e., sketched objects and binary relations between them). Hence, there are two principal methods to simplify an association graph.

The first method attempts to reduce the number of nodes in the association graph (i.e., by disregarding individual sketched objects). Filters can be used to isolate less important objects, which the system can exclude from further processing. These filters can be based on such parameters as object distribution, heuristics, or context. The efficiency considering the number of components in the association graph increases by $O(n^2)$, because for every object eliminated there are exactly n components less in the graph ($n-1$ edges one node). However, by disregarding a specific set of objects one might accidentally exclude objects that are crucial for the meaning of the sketch.

The second approach to simplify a complete association graph is to drop specific edges, that is, ignoring specific binary relations. Reducing the number of binary relations decreases the number of model components by $O(n)$. This method allows for a selective elimination of components, while retaining the full set of sketched objects and nodes in the association graph.

Although both approaches simplify an association graph, it is questionable to omit sketched objects at the time when the symbolic model is generated. An exclusion of specific sketched objects from the association graph may be appropriate if, for instance, the system has to reprocess a simplified version of the digital sketch. To the initial configuration of the association graph, however, it appears more adequate if *all sketched objects* and a selected *subset of all binary spatial relations* are considered. The goal of such a selection process is to reduce only those binary relations that are implied by the configuration of other relations or irrelevant to the semantics of a sketch.

5.3.1 Methods to Simplify the Association Graph

The reduction of the number of binary relations in a sketch requires a classification scheme that determines if relations are essential or not. The relevance of binary relations between sketched objects can be expressed in regard to *context*, *spatial location*, or *temporal sequence* (Blaser 1998). A relation is essential if at least one classification method considers the relation important.

- *Semantics and Context:* A highly evolved method of interpreting the relevance of object-object relations in a sketched scene is taking into account the semantics of objects and the context of the sketch. An interpretation at this *conceptual level* requires semantics knowledge about the sketched objects and their spatial configuration. How people interpret spatial scenes is yet unknown (Laurini and Thompson 1992), however, it is likely that this is done primarily based on perceived concepts.

- *Spatial Distribution:* Another method to evaluate binary relations between objects relies solely on geometrical aspects of a sketched scene. Interpreting a sketch at this *geometrical level* and applying Tobler's First Law of Geography (Tobler 1970) allows the classification of object pairs according to their spatial closeness. This approach is well suited for an automatic evaluation of a sketch, because it requires only knowledge about the location and geometry of objects, but not about their semantics.

- *Temporal Sequence:* Sketching is a sequential process and sketched objects are typically drawn one after another (Blaser 1998). Based on findings about the sequence of objects in spatial sketches it is possible to classify object pairs with respect to their temporal closeness. An evaluation of binary relations on this *temporal level* is computationally inexpensive, because the number of additional relations is at most the number of objects in a sketch minus one.

5.3.1.1 Object Neighborhood

Each of these classification methods leads to a specific association graph. Depending on the approach, nodes can be connected (through edges) to a variable number of other nodes. Nodes with no connection to other nodes in the graph are called disconnected; the maximum number of connections per node is `n-1`. Using the term *neighborhood* for all nodes that are directly connected to a particular node and taking into account that every node represents an individual sketched object, one can distinguish between three concepts that describe the connectedness of an object with its environment:

- Contextual Object Neighborhood,
- Spatial Object Neighborhood, and
- Temporal Object Neighborhood.

The sketch in Figure 5.7a is intended to support a verbal description of a person explaining to another person the way from a house to the railway station. The arrows in Figure 5.7b link objects spatially, considering only direct neighbors. In Figure 5.7c the sketched objects are connected according to their temporal drawing sequence, whereas in Figure 5.7d objects are linked based on a possible conceptual interpretation.

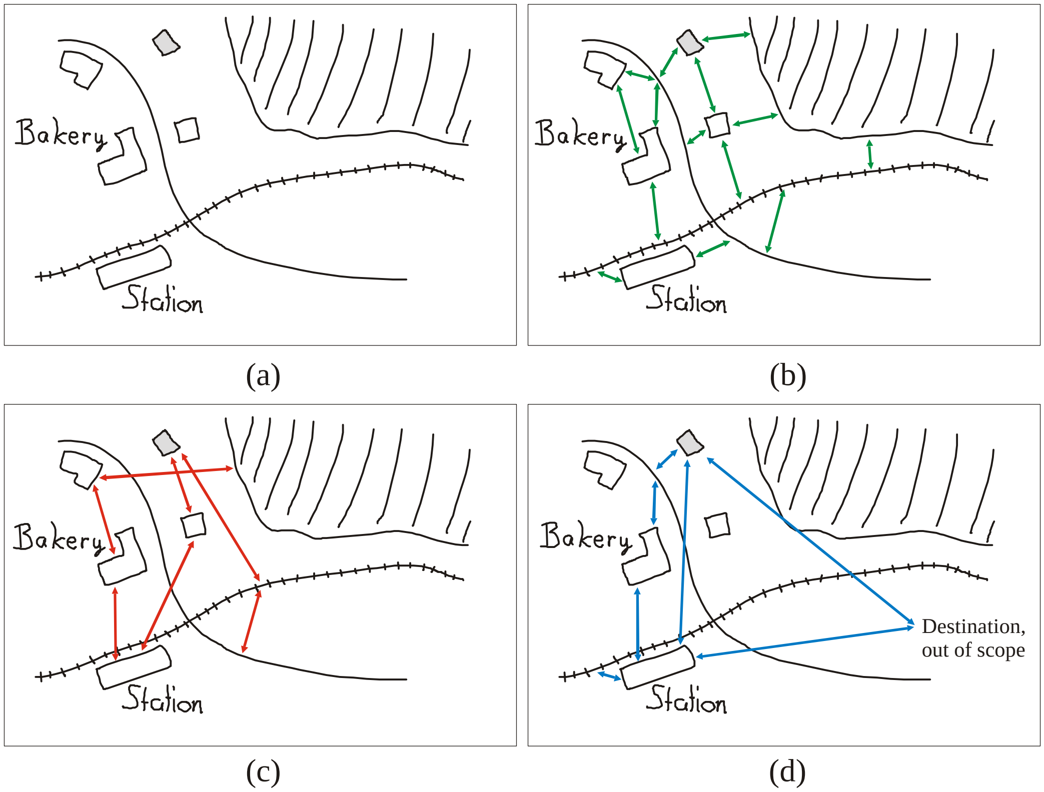


Figure 5.7 (a) The original sketch and (b) spatially, (c) temporally, and (d) contextually derived binary relations between sketched objects.

The sketch in Figure 5.7a contains eight distinct objects with a total of 28 possible binary object-object relations. There are 14 spatial, seven temporal, and seven contextual binary relations. Their union, without redundant relations, comprises 21 relations, which is 75% of the total number of possible binary relations in this configuration. For larger object configurations this ratio is much smaller. The number of possible binary relations is coupled with the number of objects and grows by $O(n^2)$ (Section 5.2.1). Conversely, the number of relations that is derived from temporally linked objects grows linearly by $O(n)$. The size of the set of relations that results from objects considering their spatial closeness depends on the selected classification method; however, it can be expected that this size is considerably smaller than that of the complete set of relations, because only relations to objects in the close vicinity of objects are being considered.

The evaluation of the number of binary relations on the conceptual level is more difficult, because this number depends on people's individual approaches of interpreting spatial scenes. However, research in cognitive science (Lakoff and Johnson 1980) suggests that not everything is mentally linked with everything else, indicating an increase of binary relations by less than $O(n^2)$ when objects are added for the conceptual level.

5.3.1.2 Evaluation of the Simplification Methods

Each of the three approaches extracts a specific subset from the complete set of binary relations. These subsets can be considered separately or they can be combined. To evaluate individual simplification methods concerning their suitability to generate a representative network of binary relations it is necessary to check their association graphs against a set of requirements.

- *Complete coverage:* All objects must be interlinked so that every object remains connected (i.e., the association graph may not have any disconnected nodes).
- *Representativeness:* The selected set of relations should include the relevant binary relations in a sketch.
- *Balancedness:* The association graph must be well balanced, that is, the number of edges per node (relations per objects) must be such that each node is sufficiently connected to the rest of the graph; however, redundancy should be avoided.
- *Cognitive feasibility:* The reduced association graph must be cognitively feasible, that is, the resulting binary relations between sketched objects should be intuitive.
- *Implementational feasibility:* The implementation of algorithms that are required to obtain the reduced association graph must be efficient.

An association graph that is created based on the spatial neighborhood of objects is a balanced representation of a sketched scene. All objects are interconnected and, because of Tobler's First Law of Geography, spatial neighbors share a mental connection. Research in computational geometry (Preparata and Shamos 1985; O'Rourke 1993) suggests a number of robust algorithms that can be used to efficiently compute different spatial neighborhoods of objects in a spatial scene.

A context-derived association graph represents a sketch and its elements according to the interpreting person's intuition. In general, this results in a network of objects and binary relations with a high expressiveness, because only relevant object-object connections are represented. However, certain objects may have no relation that connects them to the network, leading to an unbalanced association graph with potential "holes" (i.e., nodes that are not connected to the graph). An automatic generation of the context-derived association graph is also problematic, because extensive knowledge about the sketched objects and the context of the sketch is required. This is an important difference to the approach based on the spatial neighborhood.

Association graphs that are based on the temporal drawing sequence of sketched objects are the most simple to generate. However, they are unable to represent a sketch with the same accuracy as the two other graphs, because each object is connected with at most two other objects–its predecessor and its successor. Therefore, the temporal graph is not suitable for representing a sketch alone. If used as a complementary graph, however, it may capture important relations as well (Blaser 1998).

The concept of the spatial neighborhood of sketched objects provides a combination of simplicity, balancedness, and efficiency that it not matched by any of the other approaches interconnecting spatial objects in a sketch. We focus, therefore, on spatial methods to create a reduced association graph. The temporal sequence approach is excluded so that spatial scenes without any temporal information can be processed as well. The context-based approach requires additional semantic knowledge, which is currently not available. The interpretation and simplification of the association graph is, therefore, based on the sketch's geometry alone.

5.3.2 Spatial Neighborhood Graphs

Most GIS applications rely on a variety of algorithms that describe the vicinity of spatial entities for operations, such as finding the closest neighbor of an object or generating buffer zones. Depending on the requirements of an application, different approaches have been established. Important graph structures that take into account the spatial distribution of objects are the Minimum Spanning Tree (Toussaint 1980), the Relative Neighborhood Graph (Lankford 1969; Toussaint 1980; Jaromczyk and Toussaint 1992), and the Gabriel Graph (Gabriel and Sokal 1969; Liotta 1996).

One of the most widely used methods to connect points in space is the Delaunay Triangulation (Figure 5.8). It partitions the space between a set of points in the Euclidean plane into triangles such that no four points of this set are cocircular (O'Rourke 1993). The result is a well-balanced graph with no disconnected points. The dual of the Delaunay Triangulation is the Voronoi Diagram. It is defined such that each node of the Delaunay Triangulation is the nucleus of a specific area of the Voronoi Diagram. These areas are bounded by the perpendicular bisectors of the nucleus and the set of its neighboring points. Both graphs belong to the most fundamental data structures in the field of computational geometry (Aurenhammer 1991) and a large number of natural phenomena can be modeled using either approach.

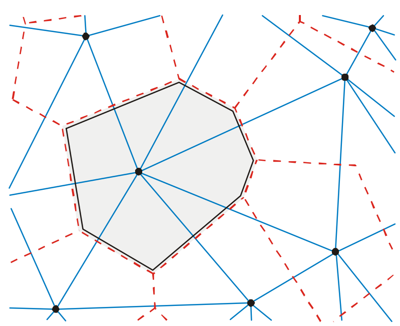


Figure 5.8 Voronoi Diagram (dashed lines) and Delaunay Triangulation.

Figure 5.8 shows a section of a Voronoi Diagram and its Delaunay Triangulation. The set of points is connected through a Delaunay Triangulation. The shaded area comprises all points that are closer to the nucleus of this region than to any other point. The boundaries of that region (dashed lines) are part of the Voronoi Diagram. The concept of the Delaunay Triangulation can be extended so that lines and regions can be connected as well. For this purpose the vertices of an object's outline (line or region object) are considered nodes and the edges of the outline are considered *constrained edges* of the Delaunay Triangulation (i.e., the requirement for cocircularity is dropped for constrained edged).

5.3.3 The Reduced Association Graph

A typical Delaunay Triangulation is based on a set of points. A sketch, however, typically consist of points, lines, and regions. To take into account this characteristic, it is necessary to introduce the outline of sketched objects as constrained edges (Section 5.2.3). A second issue to consider is the actual purpose of building an association graph, which is to link spatial neighboring objects with each other. To qualify as neighbors, two objects must share one or more Voronoi Edges. Because of the dual characteristic of the Voronoi Diagram and the Delaunay Triangulation, a shared Voronoi Edge is the same as one or more connecting edges of the Delaunay Triangulation. Figure 5.9 illustrates a constrained Delaunay Triangulation, which captures the spatial object neighborhood in a sketch.

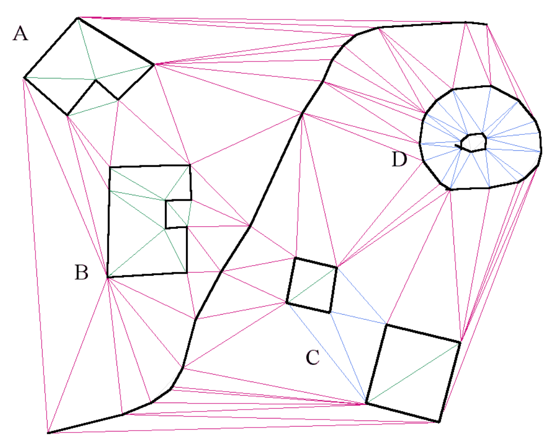


Figure 5.9 Screenshot of the Delaunay Triangulation of a sketched scene. Object C and D are composite objects, each consisting of two components.

There are three different types of edges in the example sketch in Figure 5.9: (1) edges that connect different objects (red edges), (2) edges that connect different parts of the same object (blue edges), and (3) edges that connect vertices inside the same object (green edges). In order to be spatial neighbors, two different objects need to be connected with at least one edge.

Figure 5.10 shows the association graph of the sketch from Figure 5.9 in form of a diagrammatic representation. Sketched objects are substituted by disk-shaped symbols that are connected according to their spatial neighborhood with diamonds, representing binary spatial relations.

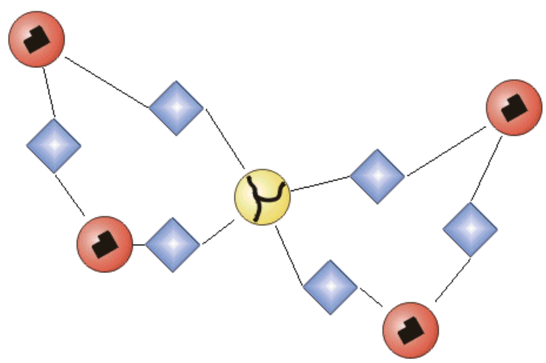


Figure 5.10 Diagrammatic representation of the sketch from Figure 5.9.

The association graph in Figure 5.10 shows five objects (four region objects and one line object) that are linked through six binary relations (diamonds). Four binary relations are not shown, because they do not establish a spatial neighborhood relation.

5.3.3.1 Qualitative Characteristics

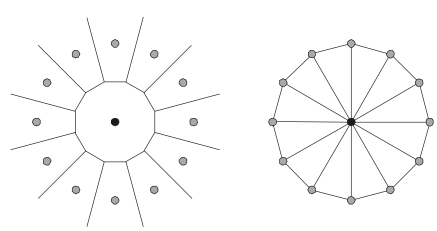
The association graph connects sketched objects in a natural way and meets the requirements postulated in Section 5.3.1. Objects are connected to their spatial neighbors; this approach has computational advantages for processing a spatial scene. For instance, if the data model stores objects *and* their neighborhood relations, then it is possible to directly answer queries, such as “which objects are immediately north of this object?" or "which object are on the right hand side of this street that leads from A to B?" Another advantage of the reduced association graph is that it can be updated efficiently. If a sketch is edited and objects are inserted or deleted, it is only necessary to update the model locally.

Large objects may have a shielding character. For instance, a long street divides a suburban settlement into houses that lie on the left hand side and on the right hand side of the street. Houses opposite to each other, however, are indirectly linked to each other.

5.3.3.2 Quantitative Characteristics

The Delaunay Triangulation and its dual the Voronoi Diagram have been investigated extensively in the literature of computational geometry (Fortune 1985; Preparata and Shamos 1985; Aurenhammer 1991; Fortune 1992; de Berg *et al.* 1997). However, in most cases researchers have focused their investigations on point sets; therefore their quantitative analysis cannot simply be adopted.

For a Voronoi Diagram consisting of `n` points it can be shown that there are at most `n-1` edges and `n-1` vertices per Voronoi Polygon (Figure 5.11a). Because the Delaunay Triangulation is the dual of the Voronoi Diagram, a point can have at most `n-1` neighboring points in a Delaunay Triangulation (Figure 5.11b). However, Figure 5.11b also shows that–besides the center point–all other points have only three neighbors.



(a) (b)

Figure 5.11 (a) A special case of a Voronoi Diagram and (b) its dual the Delaunay Triangulation.

Euler's equation (Equation 5.2) can be applied for every convex polyhedron with m_v nodes (vertices), m_f faces, and m_e edges. This includes the single unbounded region; hence for a graph in 2D-space the number of faces can be reduced by one (Equation 5.3). Every vertex in the Delaunay Triangulation has a degree of ≥3 (Equations 5.4 and 5.5).

$$m_v - m_e + m_f = 2 \quad (5.2)$$

$$m_v - m_e + m_f = 1 \quad (5.3)$$

$$m_f \leq 2 \cdot m_v - 4 \quad (5.4)$$

$$m_e \leq 3 \cdot m_v - 6 \quad (5.5)$$

If Equation 5.5 is to be used for the Voronoi Diagram, then an additional point in infinity v_∞ has to be introduced, because the Voronoi Diagram consists of edges *and* half-edges (which have only one vertex). This final point has to be subtracted for this evaluation, thus the maximum of vertices in a Voronoi Diagram is $2 \cdot m_f - 5$.

Each vertex in a Delaunay Triangulation can have at most n-1 connected edges, each edge bounds exactly two vertices, and there are at most $3 \cdot m_f - 6$ edges in the whole Delaunay Triangulation. Therefore, if all vertices of a Delaunay Triangulation are substituted with objects and edges with binary relations we can deduce for a very large graph that the *average number of neighbors* of an object is less than six.

Equations 5.2 – 5.5 are valid for Delaunay Triangulations consisting of points. A sketch, however, consists mostly of line and region objects. If sketched objects would not intersect then one could simply shrink all objects until they are mere points and apply Equations 5.2 – 5.5; however, objects in a sketch *do* intersect. Large objects can intersect with many other objects or they can have a large number of highly distributed and disconnected parts. This leads to an association graph that frequently intersects itself so that Euler's Equation loses its applicability. Figure 5.12 shows that the association graph can theoretically have a quadratic number of edges.

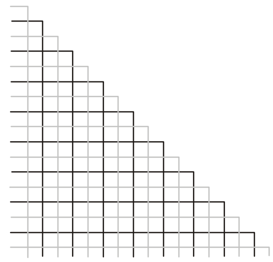
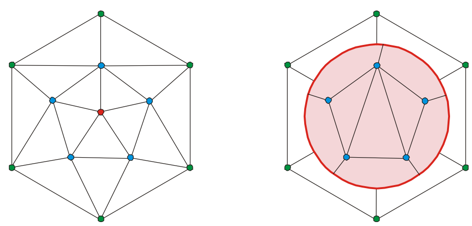


Figure 5.12 Sketch with 17 line objects, where each object intersects every other object.

An effect that leads to a decrease of binary relations in the association graph is when large objects are shielding smaller objects (Section 5.3.3.1). The two graphs in Figure 5.13 have the same number of objects, but 5.13b has a disk-shaped region object instead of the center point of the situation in 5.13a. The region object in 5.13b shields the five inner points from being connected to the six outer points, which reduces the number of links in the graph from 27 to 24. Similar observations can be made with line objects.



(a) (b)

Figure 5.13 (a) A graph consisting of 12 point objects and (b) a graph consisting of 11 point objects and one disk-shaped object.

The survey about people's sketching behavior (Section 4.4.1) indicated that only 8.2% of all possible binary relations in a typical sketch were of type non-disjoint and of these 41% (3.4% of the total) were of type *overlap*. Other non-disjoint relations, such as meet or contain do not have the same impact on the growth of the association graph.

Based on these considerations it seems reasonable to assume that a situation, such as shown in Figure 5.12, is very rare and that Equations 5.2 - 5.5 can be applied for a set of sketched objects as well. It can, therefore, be expected that the reduced association graph is a compact representation of a sketched scene and that the graph grows linearly by O(n) (Figure 5.14).

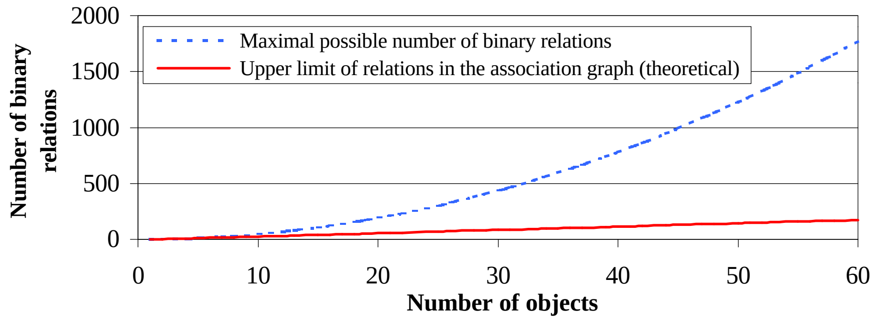


Figure 5.14 Comparison of the increase of binary relations between the complete and the reduced association graph in function to the number of objects.

Observations made during the evaluation of both approaches using a prototype implementation (Blaser 1999a) support this claim: the highest average number of object neighbors experienced in our tests was 5.96. The dataset that produced this result contained 1000 sketches, each with 24 randomly distributed and strongly overlapping objects.

The relevance of the quantitative characteristics of the reduced association graph increases with the number of objects. Therefore, if it is possible to transfer the developed methods for sketched objects to spatial objects in general, then the reduced association graph promises to store spatial relations efficiently within large spatial databases.

5.4 Extending the Digital Sketch

The association graph provides a framework for additional extensions that increase the stability and expressiveness of the digital sketch. The computational effort for measures that rely on this association graph is small, because the network itself is already computed.

5.4.1 Scene Completeness

The digital sketch consists of two types of entities: spatial objects and binary spatial relations (Section 5.2.1). To achieve a similarity of 100% between a sketched query and a retrieved database record, all components in the sketched query must have corresponding entities in the database records. If objects or relations are missing in the retrieved data record, then the scene is incomplete, which results in a reduced similarity value.

5.4.1.1 Object Completeness

Object Completeness is a measure to describe the ratio between the number of sketched objects with a corresponding object in the retrieved dataset and the total number of objects in the sketch. If all objects in the query have a counterpart in the dataset, then the value for the *Object Completeness* is 100% otherwise it is proportionally less. The simplest approach considers each object of equal importance (Equation 5.6a); a more elaborate model gives different weights to each object (Equation 5.6b).

$$ObjectCompleteness = \frac{n}{N} \quad (5.6a)$$

$$ObjectCompleteness = \frac{\sum_{i=1}^{n} O_i \cdot q_i}{\sum_{i=1}^{N} q_i} \quad (5.6b)$$

with: N = Number of objects in the sketched query

n = Number of object in the retrieved dataset that were associated with objects in the sketched query.

O_i = 1 if the i-th object in the original sketched query has a corresponding object in the retrieved dataset, 0 if not.

q_i = The weight of the i-th object in the sketched query

This evaluation of objects is equivalent to the comparison of the nodes with corresponding nodes in two association graphs. The assessment method depends only on the existence or non-existence of corresponding object pairs and is invariant under all other parameters. It can, therefore, be used for other models, such as the complete approach (i.e., the association graph considering all binary relations).

5.4.1.2 Relation Completeness

Similar to the Object Completeness, the *Relation Completeness* describes the ratio between the number of spatial relations in the sketched query and the number of corresponding spatial relations in the retrieved datasets. Instead of counting the number of binary relations (edges in the association graph) in each representation, this method takes into account only existing object connections, that is, a relation AB has to exist in the sketched query *and* the retrieved dataset to be considered. The formalization can, again, be issued for the weighted and non-weighted case (Equation 5.7a and b).

$$RelationCompleteness = \frac{m}{M} \tag{5.7a}$$

$$RelationCompleteness = \frac{\sum_{i=1}^{m} R_i \cdot p_i}{\sum_{i=1}^{M} p_i} \tag{5.7b}$$

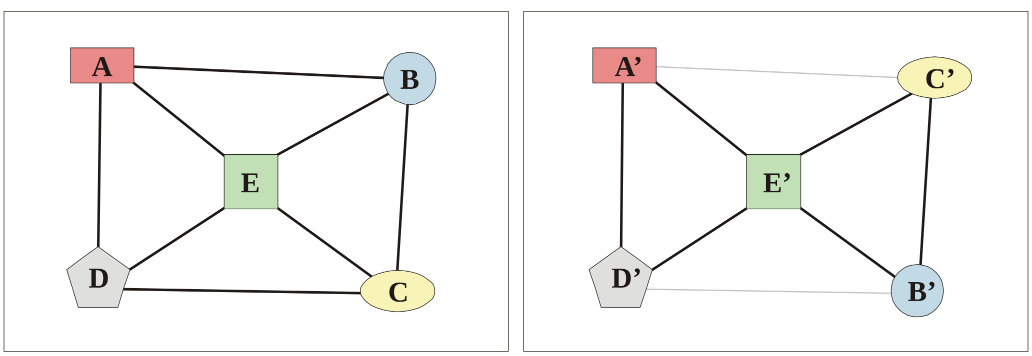
with: M = Number of selected relations in the sketched query

m = Number of corresponding relations in the retrieved dataset

R_i = **1** if the i-th relation in the original sketched query has a corresponding relation in the retrieved dataset, **0** if not.

p_i = The weight of the i-th selected relation in the sketched query

This evaluation of binary spatial relations is equivalent to the comparison of the edges with corresponding nodes and edges in two association graphs. Figure 5.15 demonstrates the evaluation of the *Relation Completeness*.



(a) (b)

Figure 5.15 The association graph of (a) the sketched query and (b) a retrieved database record; relevant binary relations are depict as thick lines.

Based on the association graph of the sketched query (Figure 5.15a) a database record of a scene is retrieved (Figure 5.15b). Although both graphs share the same number of edges, the two association graphs are not identical, because the two relations

AB and DC of the sketched query have no corresponding relations in the retrieved dataset. With 6 out of 8 common edges, Equation 5.7a yields a *Relation Completeness* of 75%.

While *Object Completeness* is independent of the association graph used, an evaluation of the *Relation Completeness* is only possible for models that are based on a subset of the complete set of binary relations.

5.4.2 Inner Orientation

The *Inner Orientation* captures the distribution of objects in space. It can be used to assess the similarity between two scenes with respect to the orientation of each scene's cells. A cell in this respect is made of any three interconnected objects and their relations. Hence, the minimal configuration for the *Inner Orientation* is three objects. Objects must not necessarily be immediate neighbors. The *Inner Orientation* can, therefore, be used for a digital sketch that is based on the complete association graph as well. However, in this case, we would have to compute a large number of triangles (Equation 5.8).

$$C_n^3 = \binom{n}{3} = \frac{n!}{3! \cdot (n-3)!} \tag{5.8}$$

with: n = Number of associated objects in the sketched query

Using the reduced association graph as a based to decide which objects are connected to each other, this number can be reduced to less than $2 \cdot n - 4$ triangles (using Equation 5.4). The method to assess the similarity of two spatial scenes according to their *Inner Orientation* is based on a comparison of the number of corresponding triangles with an equal orientation and the total number of triangles (Equations 5.9a and 5.9b).

$$InnerOrientation = \frac{s}{S} \tag{5.9a}$$

$$InnerOrientation = \frac{\sum_{i=1}^{t} s_{ABC_i} \cdot \left(\sum p_{ABC}\right)_i}{\sum_{i=1}^{t} \left(\sum p_{ABC}\right)_i} \tag{5.9b}$$

with: S = Total number of triangles in the sketched query

s = Number of triangles with a corresponding inner orientation

s_{ABC_i} = **1** if the i-th triangle in the original sketched query has the same orientation as the corresponding triangle in the retrieved dataset, **0** if not.

p_{ABC} = Weight of the three involved relations in the triangle (AB, BC, CA)

For this purpose the objects in the reference scene (i.e., the association graph of the sketched query) have to be interconnected. These interconnections (i.e., edges in the association graph) may intersect. The next step involves the construction of the corresponding network in the query scene. Subsequently, the orientations of corresponding triangles are compared. Figure 5.16 illustrates the procedure to compute the similarity of the *Inner Orientation* between two association graphs.

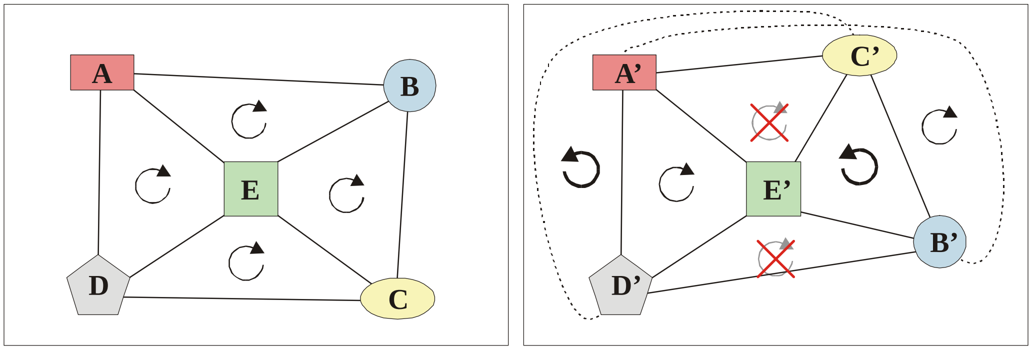


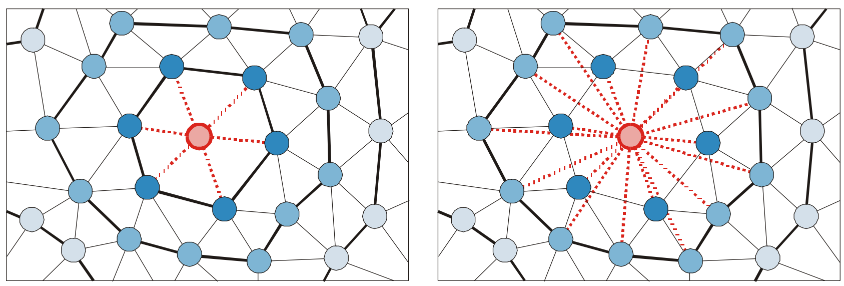
Figure 5.16 The association graph of (a) the sketched query and (b) the retrieved configuration. Rotation arrows indicate the inner orientation for each individual cell. Crossed out arrows indicate missing inner orientations in the sketched query.

Only one triangle in Figure 5.16b stays completely unchanged (A'E'D') if compared with Figure 5.16a. The triangle BCE has unambiguously changed its orientation. If the two triangles ABE and CDE have changed their orientation depends on the specific spatial configuration of the dataset. In the example in Figure 5.16b the orientation of CDE has changed, while that of ABE has not. Equation 5.9a yields a value of 50% for the similarity of the *Inner Orientation* between the two configurations.

The anchor point of an object is crucial for this evaluation and must be determined carefully. In a general case the geometric center, or the center of an object's MBR or that of its *Tilted Minimum Bounding Rectangle* (TMBR) (Blaser 1999a), can be used; however, if an object is much bigger than its neighboring objects or if the ratio between its length and width is extreme, then the use of alternative measures is appropriate. Such measures could include the segmentation or decomposition of larger objects into smaller units.

5.4.3 The n-th Voronoi Neighborhood

The model of the digital sketch considers so far only immediate neighbors of objects. However, under certain circumstances it may be beneficial to consider indirect neighbors as well (Section 5.3.3). For this purpose it is appropriate to introduce the concept of the *n-th Voronoi Neighborhood* with n 1. The 1st Voronoi Neighborhood includes all immediate neighbors of an object and is directly based on the reduced association graph (Section 5.3.3). Subsequent Voronoi Neighborhoods (e.g., the 2nd, 3rd, or 4th) incrementally add the neighbors of the object's neighbors to this set of neighboring objects. Figure 5.17 illustrates the n-th Voronoi Neighborhood for n=1 (Figure 5.17a) and n=2 (Figure 5.17b).



(a) (b)

Figure 5.17 Association graphs with the connecting links between the center object and its Voronoi Neighbors (dotted red lines) for (a) the 1st and (b) the 2nd Voronoi Neighborhood.

Increasing the number of neighbors per object increases the stability of the network, because more constraints are introduced. Such an increase also allows objects to be connected across shielding objects (Section 5.3.3.2). The average number of relations per object grows with $6 \cdot 2^{(n-1)}$ new relations per object for the n-th Voronoi Neighborhood. The total number of edges (i.e., binary spatial relations) in the association graph of the n-th Voronoi Neighborhood is less then $6 \cdot m(2^n - 1)$, with m equals the number of nodes in the graph and considering that a typical node in the reduced association graph has less than six immediate neighbor nodes (Section 5.3.3.2).

5.5 Similarity Assessment between two Sketched Scenes

The similarity assessment between two sketches scenes is based on the similarity assessment of individual components of their corresponding digital sketches (Bruns and

Egenhofer 1996). Each model component of the digital sketch (e.g., topology or metric) and each component of the sketched scene (i.e., each sketched object and each binary spatial relation) can have an individual weight. The overall similarity between two sketched scenes is called *scene similarity* and is computed as the weighted average of all involved components. The following sections describe how the individual similarity values are evaluated.

5.5.1 Influence of Individual Model Components

The default weight for all components is 100%. Setting the weight to 0% is equal to disabling a model component. Objects have one main weight associated; binary spatial relations have additional weights for each individual subcomponent (i.e., topology, metric, and direction). On the sketch level there is another weight for the completeness of a sketch (5.4.1).

5.5.2 Geometric Similarity

The geometric similarity between two scenes is based on the individual geometric similarities (Blaser 1999a) of all associated object pairs (Equation 5.10).

$$S_{Geo} = \frac{1}{\sum_{i=1}^{m} p_{Geo_i}} \cdot \sum_{i=1}^{m} p_{Geo_i} O_{Geo_i} \tag{5.10}$$

O_{Geo_i} : Geometric similarity of an associated object pair

p_{Geo_i} : Weight of a sketched object

m : Number of associated object pairs

5.5.3 Topologic Similarity

The topological similarity between two spatial scenes is computed based on the topological similarities of all binary relations as described by the association graph (Equation 5.11).

$$S_{Topo} = \frac{1}{\sum_{i=1}^{n} p_{Topo_i}} \cdot \sum_{i=1}^{n} p_{Topo_i} R_{Topo_i} \quad (5.11)$$

R_{Topo_i} : Topological similarity of an associated pair of binary relations

p_{Topo_i} : Weight of the topological component a particular relation

n : Number of binary spatial relations in the association graph

The topological similarity between two binary relations is computed according to their relative difference in the *Conceptual Neighborhood* graph (Section 5.1.1). Consequently, the topology can take values between 0% and 100%, with 25% increments. For instance, if a topological *meet* relation is compared with a relation of type *covers*, then their topological similarity is 50%.

5.5.4 Metric Similarity

For each topological relation there is a set of specific formalisms that captures its metric characteristics (Section 5.1.2). The metric similarity between two spatial relations (R_{Metr_i}) is computed using all metric refinements that apply for the specific topology of the spatial relation. The reference topology for this purpose is that of the relation of the sketched query and the weight of all formalisms is equal. The metric similarity between two spatial scenes is computed based on the metric similarities of all binary relations as described by the association graph (Equation 5.12).

$$S_{Metr} = \frac{1}{\sum_{i=1}^{n} p_{Metr_i}} \cdot \sum_{i=1}^{n} p_{Metr_i} R_{Metr_i} \qquad (5.12)$$

R_{Metr_i} : Metrical similarity of an associated pair of binary relations

p_{Metr_i} : Weight of the metrical component a particular relation

n : Number of binary spatial relations in the association graph

5.5.5 Direction Similarity

The direction similarity between two spatial scenes is computed based on the direction similarities of all binary relations as described by the association graph (Equation 5.13).

$$S_{Dir} = \frac{1}{\sum_{i=1}^{n} p_{Dir_i}} \cdot \sum_{i=1}^{n} p_{Dir} R_{Dir_i} \qquad (5.13)$$

R_{Dir_i} : Direction similarity of an associated pair of binary relations

p_{Dir_i} : Weight of the direction component a particular relation

n : Number of binary spatial relations in the association graph

The prototype divides the computation of the direction similarity into two sub-problems according to the x and y-axis (Section 5.1.3). The direction similarity is the averaged sum of the two results.

5.5.6 Scene Completeness

The *scene completeness* is a parameter that compares two spatial scenes (represented by their digital sketches) on the sketch level. The scene completeness is computed based on the weighted average of the *object completeness* and the *relation completeness* (Section 5.4.1 and Equations 5.6 and 5.7).

$$S_{Comp} = \frac{1}{p_{ObjC} + p_{RelC}} \cdot (p_{ObjC} \cdot S_{Obj} + p_{RelC} \cdot S_{Rel}) \qquad (5.14)$$

S_{Obj} : Combined object completeness between two spatial scenes

S_{Rel} : Combined relation completeness between two spatial scenes

p_{ObjC} : Weight for the object completeness

p_{RelC} : Weight for the relation completeness

The scene completeness has a limiting effect on the scene similarity: if the weight of the scene completeness is set to 100% then the weighted and averaged sum of all other similarity components (geometry, topology, metric, and direction) cannot exceed the value of the scene completeness (Equation 5.14).

5.5.7 Scene Similarity

The *scene similarity* between two spatial scenes is computed based on the weighted and averaged sum of the geometry, topology, metric, and direction components of the computational model (Equation 5.15). If the weight for the scene completeness is different from 0, then the scene similarity has to be corrected accordingly (Equation 5.16).

$$S'_{Scene} = \frac{(P_{Geo} S_{Geo} + P_{Topo} S_{Topo} + P_{Metr} S_{Metr} + P_{Dir} S_{Dir})}{(P_{Geo} + P_{Topo} + P_{Metr} + P_{Dir})} \qquad (5.15)$$

$$S_{Scene} = S'_{Scene} \cdot ((P_{Comp} S_{Comp}) + (1.0 - P_{Comp})) \qquad (5.16)$$

Similar to the weights of individual sketch components (Section 5.5.1), the weights of the computational model components (P_{Geo}, P_{Topo}, P_{Metr}, P_{Dir}, and P_{Comp}) can be set on an individual basis. The default weight for each model component is 100%; however, further research is required to determine an appropriate weight distribution for involved components (Chapter 8).

5.6 Summary

This chapter focused on the translation of a sketch into a symbolic representation that is appropriate for an automatic processing. The product of this translation is called the *digital sketch* and consists of sketched objects and their binary spatial relations. The components of the digital sketch are connected through an *association graph*, with sketched object as nodes and binary spatial relations as edges. To improve the efficiency of the association graph, our goal was to reduce the number of edges in the graph. We developed a method that connects nodes with their immediate spatial neighbors. The method to assess the spatial neighborhood of sketched objects (nodes in the association graph) is based on a constrained Delaunay Triangulation, with the outline of the sketched objects representing the constrained edges. The *reduced association graph* is obtained by interconnecting only those nodes whose corresponding objects are connected through at least one edge of the constrained Delaunay Triangulation. Two objects whose nodes are connected are called *Voronoi Neighbors*. The *complete association graph*, which connects each object in the sketch with all n-1 objects, grows by $O(n^2)$. The size of the reduced association graph, however, increases only by O(n). The reduced association graph also serves as a framework for three additional extensions of the model of the digital sketch, each increasing the stability of the model.

The similarity assessment between two sketched scenes is based on the similarity assessment of individual components of their corresponding digital sketches. This evaluation involves all components of the digital sketches, that is, topology, metric, and direction of binary spatial relations, the geometry of sketched objects, and object- and relation completeness of a sketch. Each component can have an individual weight. The result of this evaluation is the *scene similarity* between two sketched scenes.

Chapter 6

Prototype of a Sketch-Based Query User Interface for GIS

This chapter gives an overview of the design and the implementation of a prototype of a sketch-based user interface to query spatial information in a GIS environment. The prototype serves as a proof of concept for Spatial-Query-by-Sketch (Egenhofer 1996b) and it is used as a test bed for the evaluation of different strategies for processing sketched spatial queries. The theoretic foundation of the implementation is based on the model of the digital sketch (Chapter 5). The following sections describe the functionality of the user interface, the implementation of principal programming classes, and the process of drawing and processing a sketched query.

6.1 Classes of the Digital Sketch

A sketched scene can be implemented as a software system using three conceptual building blocks: *sketch*, *sketched object*, and *binary spatial relation* (Section 5.2.1). Each component is implemented as a class. Figure 6.1 shows the relation between the main programming classes of the prototype implementation.

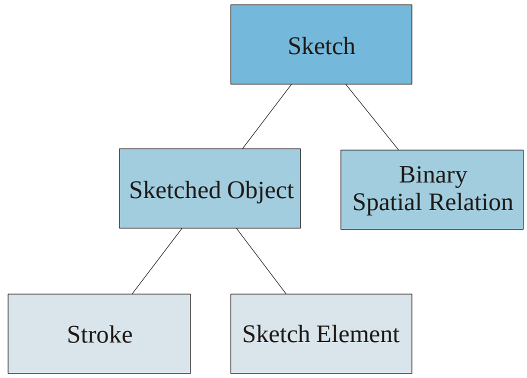


Figure 6.1 Main classes of the implementation.

Two additional classes (*stroke* and *sketch element*) were introduced to describe sketched objects at a lower level. The following sections outline the most important data members, member functions, and the purpose and role of these classes.

6.1.1 *Stroke and Point*

A *stroke* (CStroke class) is an unintelligent polygon that is created when the user draws with virtual ink on electronic paper. It is complete or closed when the user lifts the pen from the drawing surface. Strokes are responsible for storing the original geometric and temporal information of the pen movement. This raw information is recorded as a list of points (Figure 6.2). The frequency with that individual points are created depends on the user's drawing speed, the type of the input device, and the computer's performance. An individual stroke is that polygon, which is obtained by connecting the set of subsequently produced points with straight-line segments.

```
struct PPoint
{
   CPoint  Point;                // Coordinates of input point x/y
   CPTime  Time;                 // Time when point was created t
   int     PenSize;              // Pressure dependent PenSize
   int     Flag;                 // Point specific flag
};
```

Figure 6.2 Definition of the PPoint structure.

Other data members of the stroke class contain variables that are computed during the stroke assessment phase (Section 6.3.1). They reflect particular characteristics of strokes, such as a stroke's length or the number of times that the stroke changes its direction. The interface of a stroke consists of functions that are used for the stroke analysis and functions that are used for geometric operations, such as an intersection with another stroke or the computation of the distance to a point.

6.1.2 *Sketch Element*

Sketch elements are the intelligent form of strokes, that is, they have a type (either region, line, or point) and some higher order functionality. The base class for all sketch elements is CSElement. Derived from this class are CSRegion, CSLine, and CSPoint objects. Sketch elements are typically deduced from line strokes during the object assessment and interpretation phase (Section 6.3.2). The information stored in a sketch element is purely geometric (Figure 6.3). Sketch elements are usually stored as local data members of sketch objects. This is different from strokes, which are created on the heap and linked to objects by pointers.

Figure 6.3 The CSElement class is the base class for region, line, and point classes.

Because of their geometric notion, objects of the CSElement class can be used for multiple purposes, for instance, to store the outline of sketched objects, or to store intermediate results or non-visible geometric structures, such as the convex hull of a sketched object.

6.1.3 *Sketched Object*

Sketched objects are the primary building blocks of a sketch. They have a unique identity with their own functionality and data space. Sketched objects are functionally autonomous units that have some knowledge about their spatial neighborhood (Section 5.3.2). Figure 6.4 shows the most important components of the CSObject class.

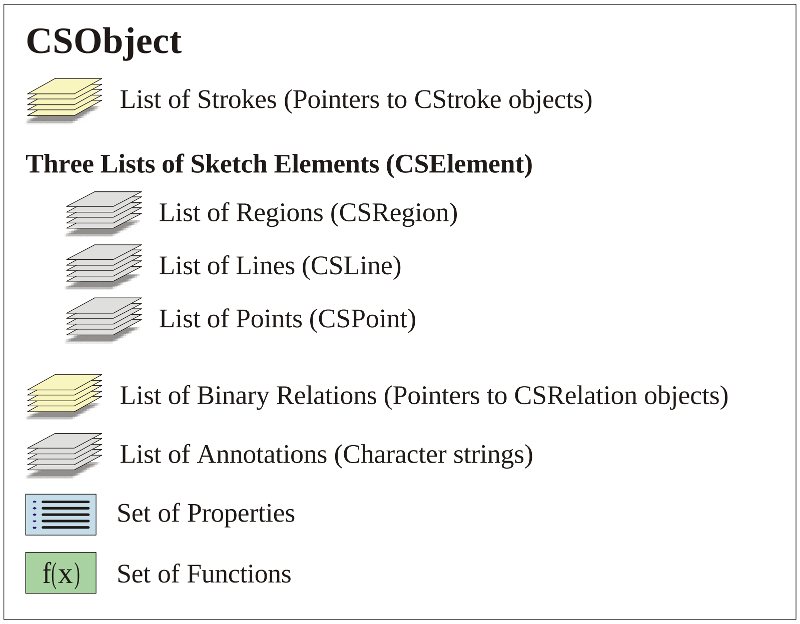


Figure 6.4 Overview of the CSObject class.

A sketched object may consist of multiple region and line elements, but it may only have one overall type (i.e., line or region). For instance, a user can draw a town with multiple boxes for houses and lines for a virtual street system. However, the prototype simplifies the sketched object (town) and stores it as a single region.

Initially, a sketched object consists of a set of strokes. These strokes are created on the heap and each stroke is affiliated with exactly one sketched object. The sketched object maintains a list of pointers to all affiliated strokes. Two other lists store all line and region elements that are associated to the object. Sketched objects can be annotated. Annotations are stored as character strings and multiple annotations per sketched object are possible. Another list stores links to neighboring objects. These links are established through pointers to binary relations (Section 6.1.4).

The interface of the CSObject class allows objects of other classes to retrieve information, update specific data members, and initiate processes. A sketched object can, for instance, draw itself as a set of strokes, an outline, or an icon. Other functions address processing specific tasks, such as the *UpdateGeometry()* function, which reevaluates all geometric parameters of an object.

6.1.4 Binary Spatial Relation

Binary spatial relations (CSBinRelation) are implemented as autonomous entities within a sketch, which is conceptually similar to sketched objects (Figure 6.5). The major difference between sketched objects and binary relations is that objects are drawn, whereas relations are not. That is, binary relations are in general not explicitly specified by the user, but derived based on the spatial configuration of the sketch (Section 5.3.3).

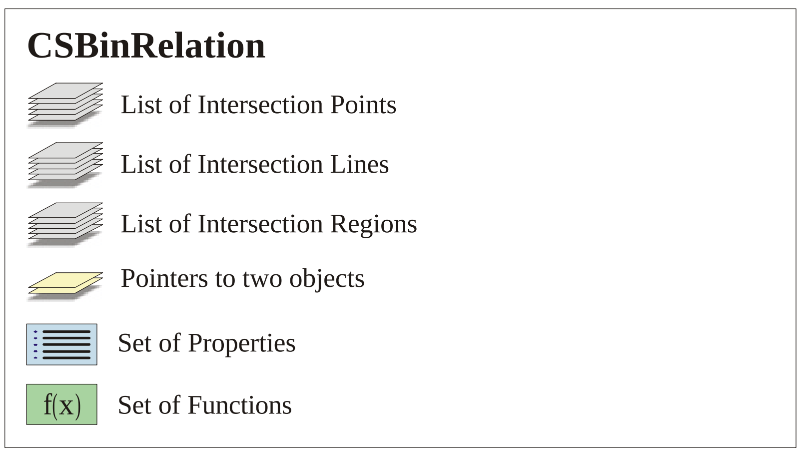


Figure 6.5 Overview of the CSBinRelation class.

Binary spatial relations are generated automatically when a sketched query is processed. Once created, relations compute their characterizing properties (i.e., topology, metric, and direction). Each relation object stores its properties as local data members and the links to involved objects as a list of pointers.

6.1.5 Sketch

The *sketch* (CSketchoDoc) is the principal data storage class (Figure 6.6). There is only one object of this class per spatial scene (sketch); however, a user may work on multiple sketches simultaneously. In Microsoft Foundation Classes (MFC) terminology a sketch is a document, derived from the MFC CDocument class. To reduce the confusion between a *sketched object* and an *object derived from the sketch class*, the latter is referred to as *document* or simply *sketch*.

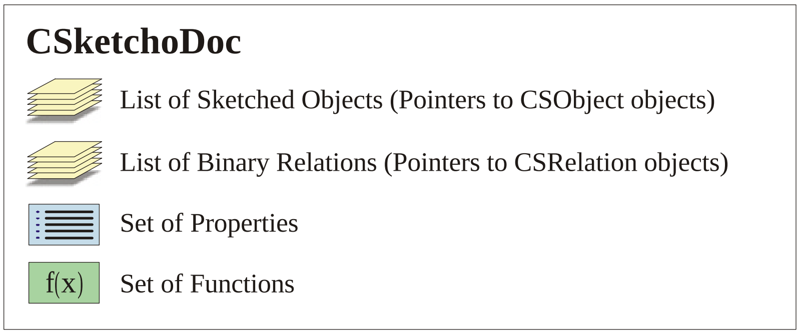


Figure 6.6 The CSketchoDoc class.

The document stores and manages two lists of pointers to sketched objects and binary spatial relations. It keeps track of their histories and is responsible for creating, deleting, storing, and retrieving sketched objects. The document plays, therefore, a central role as a platform to exchange information and initiate processes.

6.1.6 Implementation Issues

The prototype was implemented sequentially, starting with the design of the user interface and concluding with the development of tools for the query result presentation. The prototype is implemented in an object-oriented environment in C++ and C. The primary platform is a standard PC running Microsoft Windows 95/98 or NT. The graphical functions rely on the Microsoft Foundation Class library (MFC) (Microsoft 1999). Most other classes and functions have been developed specifically for this project.

6.2 User Interface

The user interface of the prototype includes a set of standard as well as sketch-specific tools to draw and interact with a sketch. The following sections describe the components of the user interface and their functionalities.

6.2.1 User Interface Metaphor

The design of the user interface of Spatial-Query-by-Sketch is based on a sketchpad metaphor (Kuhn and Frank 1991), mimicking the principal functionalities of a pen and a piece of paper. It enhances the analog sketching behavior with non-destructive editing (e.g., reposition a sketched object without the need to delete and redraw it), multiple views of the sketch (i.e., the graphical view of the actual sketch vs. an interpreted view of meaningful objects vs. a diagrammatic representation suitable for database query processing), sketch analysis tools, and polymorphous characteristics of the input device (e.g., drawing and editing with the same device).

6.2.2 Use of a Pen

A critical aspect for a smooth interaction is that the sketch pen can be used for two purposes: (1) the drawing of the query scene and (2) the interaction with the drawing to accommodate such operations as selecting drawn objects, repositioning or erasing objects, and changing the user's view over the drawing. Today's pen technology supports the selection of multiple interaction modes. The pen used to interact with the user interface of the prototype has a rocker switch (Figure 6.7) with two positions: *forward* changes the mode of the currently selected tool, while the *backward* position initiates a context-dependent menu.

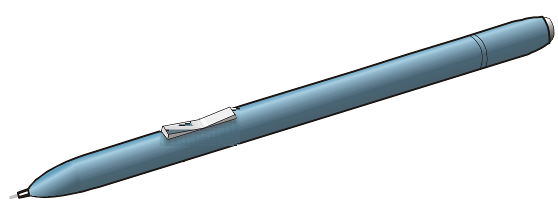


Figure 6.7 Pen with a rocker switch.

The pen tip introduces another two states (*pressed* and *not pressed*), which yields a total of $2 \cdot 3$ pen modes and enables the selection of graphical gestures during sketching and editing (Table 6.1).

Rocker switch	Pen tip	Action
Neutral	pressed	Draw, zoom, or rectangular selection
Neutral	not pressed	Move cursor
Forward	pressed	Draw straight line, container selection, move selection, pan
Forward	not pressed	None
Backward	pressed	Activate context-dependent menu
Backward	not pressed	None

Table 6.1 Pen modes.

6.2.3 Traditional User Interface Components

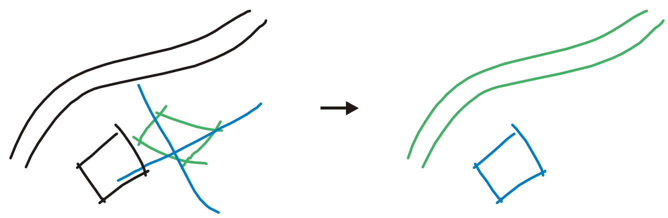
An ideal user interface of a sketch-based system consists primarily of real-estate to draw a sketch; however, for certain operations it is necessary to introduce graphic elements, such as buttons, toolbars, or dialog boxes, because this seems to be the easiest way to communicate with the system. The user interface of the prototype tries to minimize an interaction with such traditional user interface components and focuses on a direct interaction with a pen instead. For instance, to draw and query a simple sketch it is only necessary to press one single button.

6.2.4 Alternative User Interface Components

The primary goal of alternative user interface components is to support the user and enhance the communication between user and computer without being distracting (Blaser *et al.* 2000). A promising approach is to copy from techniques that people use to communicate with each other. A widely used form of interaction, in this context, is gesturing (Lipscomb 1991; Rubine 1991). The prototype understands three typical gestures.

6.2.4.1 Delete Gesture

The delete gesture is used to eliminate a previously drawn object from a sketch, putting it on the undo stack. The delete gesture consists of two crossing strokes that intersect at an angle of approximately 90° and that are of similar length. The affected sketched object is determined based on the gesture's location and on a temporal factor, that is, if there is more that one candidate, then only the most recently drawn object is eliminated. In the example in Figure 6.8 the user draws a delete gesture over a misplaced symbol of a house (a) upon which the house and the delete gesture disappear (b).

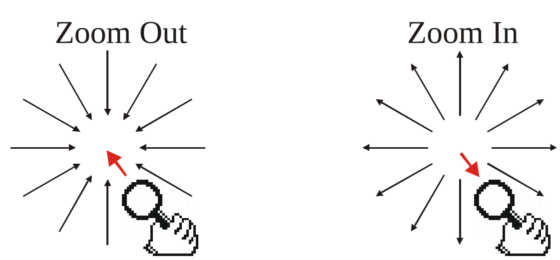


(a) (b)

Figure 6.8 Delete gesture (a) drawn over an object and (b) the resulting sketch.

6.2.4.2 Intuitive Pan and Zoom

Pen-based interaction supports a more direct zooming method than the common tool-based zooming in today's user interfaces. With the rocker switch in the neutral position and the pen tip pressed, a pen movement towards the center of the drawing area makes the user interface zoom gradually out (Figure 6.9 a), while the reverse gesture zooms in (Figure 6.9b).



(a) (b)

Figure 6.9 Zoom gesture: (a) Zooming out and (b) zooming in.

Without changing the tool, users can also pan the entire sketch into any direction they want. For this purpose, they press the rocker switch forward while the pen is in zoom mode, and drag the sketch into the desired direction.

6.2.4.3 Container Selection

Sketched elements can be selected the same way as objects are drawn; that is, the user draws a container gesture over the portion of the sketch (involving single or multiple line strokes or objects) that he or she wants to select (Figure 6.10). This method is less constrained than selecting an element with a bounding rectangle, because it eliminates the requirement of convex polygons, allowing users to specify arbitrarily-shaped areas.

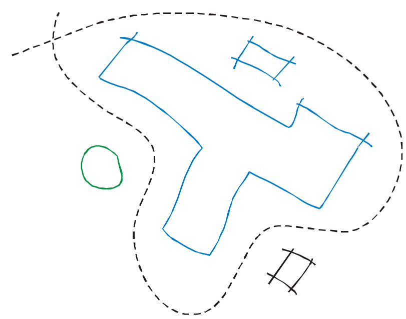


Figure 6.10 Container gesture to select two sketched objects.

Alternatively to the container selection, objects and line strokes can be selected by simply pointing at them. This allows a user to pick the appropriate method according to the actual task and his or her preference.

6.2.5 Visual Clues

The communication from the computer to the user relies solely on visual interaction. The prototype uses various visual clues, such as color or symbols, to inform the user about the state of sketched elements and the sketch.

6.2.5.1 Use of colors

If used appropriately, colors are a very powerful way to convey information (Imhof 1982). The prototype uses colors to inform a user about the current status of the sketch and its objects (Table 6.2). When an object is created it is blue, the color, which highlights the *current* or *selected* object. The object keeps this color until the first stroke of the next object—the new current object—has been drawn. At this point the object changes its color to green, marking it as the *previous* object. When yet another object is drawn then the object's color turns to black, the *consolidated* color. Other colors used include purple for detected text objects and light blue for line strokes that have been selected within the current object. With this approach user are constantly aware of how individual strokes were aggregated to objects.

Color	Semantic
Blue	Current or selected object
Green	Object that was created previous to the current object
Black	All objects that are not blue or green
Light Blue	Selected stroke within an object, or multiple selected objects
Purple	Detected text object

Table 6.2 Color codes for sketched objects.

The status of the sketch is indicated by three lights with different colors, located in the lower right corner of the user-interface (Table 6.3). The lights indicate the status of the sketched objects, the association graph, and the spatial query.

Color	Semantic
Red	Unprocessed
Yellow	Partially processed
Green	Processed

Table 6.3 Color codes for the three processing indication lights.

6.2.5.2 Cursor Icons

The user interface indicates the functionality of the selected tool with self-explanatory cursor icons. This technique is used by many modern applications and its concept has proved to be intuitive. Figure 6.11 depicts the set of implemented cursor icons according to their importance for a typical user interaction.

Figure 6.11 Cursor icons according to their importance and functionality from left to right: sketch, select, grab, zoom, pan, explicit handwriting, and typing.

6.2.5.3 Object Shadows

Shadows of objects or strokes are used when sketched elements are moved or rotated. A shadow is a light gray ghost of the object or stroke at its original location and before its modification. This technique supports the user during the process of editing, because the reference to the original location is maintained while a new spatial configuration is tested.

6.2.6 Levels of Abstraction

The prototype allows a user to display a sketch at three different levels of abstraction: (1) the original sketch with the user's line strokes; (2) the interpreted object view, which displays how the prototype translated the line strokes into distinguishable objects; and (3) a diagrammatic view, which captures the spatial relations among identified objects that are considered for query processing.

The *sketch view* accommodates the sketching environment of the user interface. It is the place where a sketch is initially created and edited, and from where a spatial query can be initiated (Figure 6.12). The sketch view is the only mandatory view.

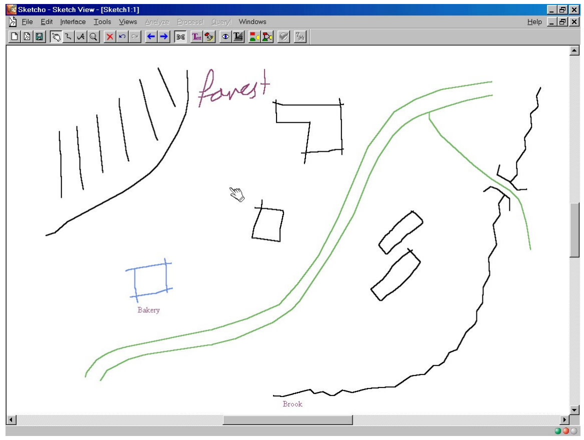


Figure 6.12 Sketch view with a sketched scene.

The *object view* displays how the system has aggregated strokes into objects (Figure 6.13). This view shows the simplified and interpreted outline of each object. It distinguishes different object types (lines, regions, and symbols) through the use of different colors (Section 6.2.5.1). The object view allows a user to evaluate the interpreted sketch and modify it if desired.

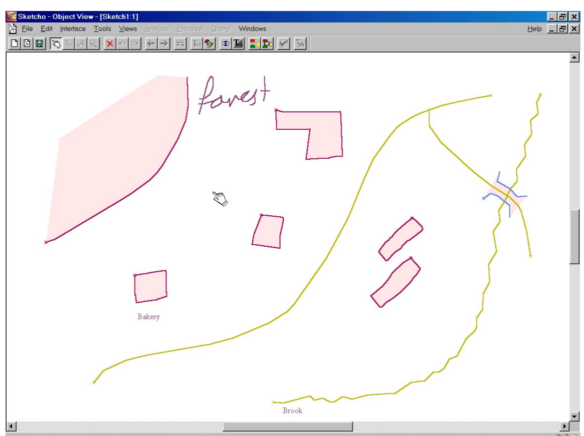


Figure 6.13 Object view of the sketched scene in Figure 6.12.

The *diagram view* is the most abstract view of a sketch (Figure 6.14) as it displays objects and binary spatial relations symbolically. Spatial relations are generated and computed according to the configuration of the digital sketch, while the set of objects is given by the sketched query. This view allows a user to examine and edit certain aspects of a sketch better than in the sketch view or the object view. For instance, objects or relations can be selected (i.e., enabled) or un-selected (i.e., disabled) for query processing and binary spatial relations can be created, deleted, and modified by manipulating their relation icons. The latter is particularly useful for editing parts of a sketch that were drawn imprecisely.

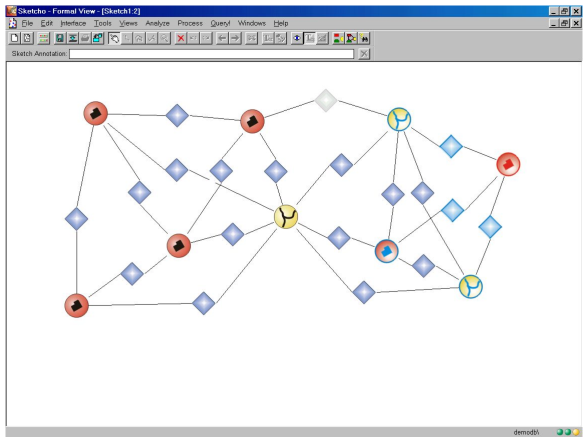


Figure 6.14 Diagram view of the sketched scene in Figure 6.12.

The user’s guide (Blaser 1999b) and the technical report (Blaser 1999a) of Spatial-Query-by-Sketch describe other functions and features of the individual views in more detail.

6.3 Sketch Processing

Drawing a sketch is a sequential procedure (Section 3.1.1). The interpretation of a sketch can be implemented accordingly (Figure 6.15). This approach is different from other data interpretation tasks in computer science, such as photogrammetric feature extraction or optical character recognition, where applications frequently have to interpret snapshot-like data.

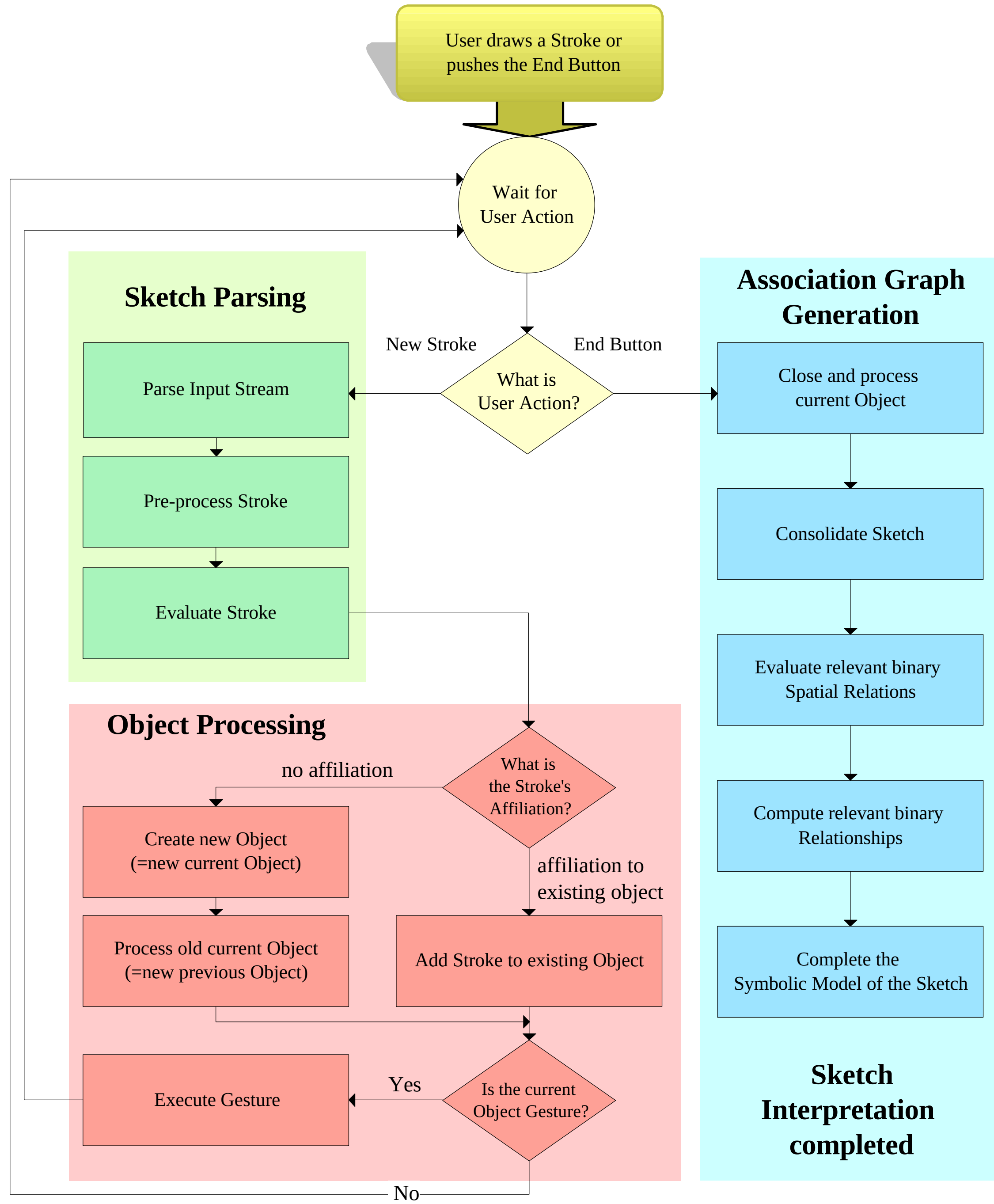


Figure 6.15 Sketch-processing loop.

The sketch-processing loop consists of three phases: *sketch parsing*, *object processing*, and *association graph generation*.

6.3.1 Sketch Parsing

The sketch parsing process consists of a first quality check of retrieved strokes combined with a simplification method. This approach reduces the number of points, while retaining the original shape of a stroke. Subsequently the stroke is evaluated and an attempt is made to associate the stroke with a previously drawn object.

6.3.1.1 Line Stroke Simplification

The number of points in a freehand line stroke depends on the user's drawing speed, the type of the input device, and the computer's performance (Section 6.1.1). While in drawing mode, the application records the pen's location. This process yields a high-resolution line that captures intended direction changes as well as small, unintended changes in the line's direction. For query processing, however, such sketched strokes need to be simplified. The prototype uses the Douglas-Peucker Algorithm (Douglas and Peucker 1973) to filter significant breakpoints (Figure 6.16). This approach uses a single tolerance value and eliminates unnecessary points based on how far these points are from a generalized line shape.

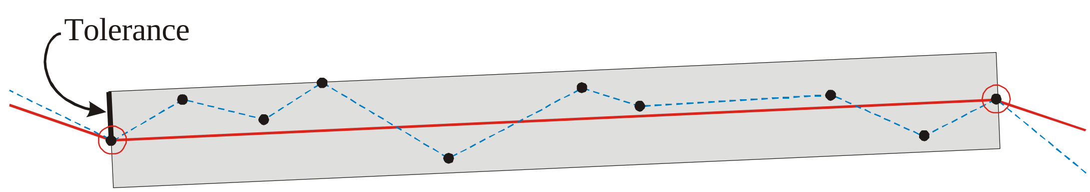


Figure 6.16 Principle of the Douglas-Peucker Simplification Algorithm.

The dashed fine line in Figure 6.16 is the original line stroke, while the red line is the result of the simplification. The major incentive for reducing the number of points in lines is that polygons with fewer points focus on the essence and, therefore, are easier and more efficient to process. The tolerance value is set by the user and should be configured such that the sketch retains its character.

6.3.1.2 Line Stroke Sequencing

After a stroke has been preprocessed, it must be aggregated into an object. For each new stroke there are three cases, distinguishing to what object the new stroke may belong:

- the current object,
- an object other than the current one, or
- no other previously drawn object (i.e., the line stroke is the first of a new object).

The criteria for this distinction are based on the location of the new stroke in relation to other objects and on the time difference (δt) between the previously drawn stroke and the current one. Each measure relies on thresholds that can be adjusted according to personal preferences. Figure 6.17 shows a sample distribution of the connectivity function for the time δt between two strokes. In this example, if δt is less than 1.2 seconds then the stroke will be associated to the previously drawn object. After that threshold the connection value decreases linearly as a function of time.

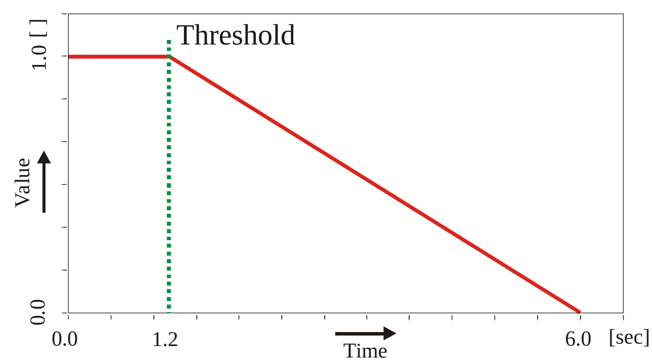


Figure 6.17 Relation of the time interval δt between two strokes and the connectivity value.

The model for the spatial sequencing of strokes is based on two factors:

- The closeness between the boundary points of the new stroke and the boundary points of a previously drawn object. If the distance between these points is smaller than or equal to a set threshold, the gap between the points is closed and the latest line stroke is aggregated to the object.

- One of the line stroke's boundary points is close to the edge of another object. Again a distance threshold is employed to determine if the stroke belongs to the object and whether to close an eventual gap, or whether to eliminate an overshoot.

The process of spatial sequencing line strokes makes use of a buffer zone around the stroke and its boundaries (Figure 6.18). Any line stroke drawn later whose boundary falls within the buffer zone gets connected to the first stroke (lines C and D in Figure 6.18), while lines stay disconnected as long as they intersect the buffer zone without starting or ending inside the buffer zone (lines A and B in Figure 6.18).

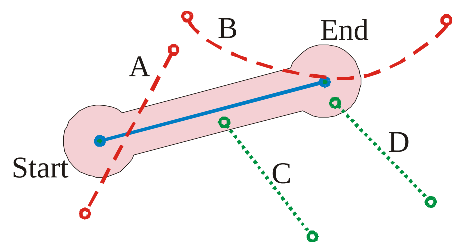


Figure 6.18 Spatial sequencing of line strokes.

The prototype gives users feedback about the sequencing of strokes by dynamically coloring the drawn objects (Section 6.2.5.1). This method highlights how strokes are aggregated to objects and helps a user to quickly identify strokes that have been associated with the wrong object. Regrouping, re-associating, and modifying objects can be done at any time. The prototype provides the following editing tools:

- Attach or detach a stroke or groups of strokes to or from an object.
- Break objects apart into single strokes (ungroup).
- Group strokes or objects to form a composite object.
- Delete strokes or objects.
- Move or rotate objects.
- Copy, paste, and duplicate objects.

This set of functions is sufficient for a sketch-based user interface for GIS, according to findings during human subject testing (Blaser 1998).

6.3.2 *Object Processing*

Objects are automatically processed when they are complete, that is, when the most recently drawn line stroke was associated to a new object. Already processed objects are re-processed when associated strokes are added, moved, or deleted. Object processing transforms the set of line strokes of a sketched object into geometric figures suitable for query processing. This task involves clean-up operations of line strokes, the extraction of object characteristics, the simplification and interpretation of objects, and the distinction between region objects, line objects, and symbolic gestures.

6.3.2.1 Text Detection

The prototype allows a user to annotate sketched objects by typing or writing. Typing is straightforward and the only concern is the correct association with a sketched object. Handwritten annotations are more complex, because the system has to identify what is a sketched object and what is handwritten text. The prototype uses a set of heuristic tests to detect handwritten annotation. The tests are based on observations of human subjects (Blaser 1998) and consider the following characteristics of line strokes: main direction, curvature, number of abrupt direction changes, and the accumulated directional change. Typed or detected handwritten annotations are associated to sketched objects according to their spatial location. Handwritten annotations are not parsed and translated into ASCII text.

6.3.2.2 Segmentation

Segmentation is the process of breaking the set of strokes of an object into non-intersecting line segments. During this process, strokes can be divided or marginally extended (Figure 6.19). The segmentation of an object is a central task, because subsequent processes rely on the set of segments rather than on original strokes.

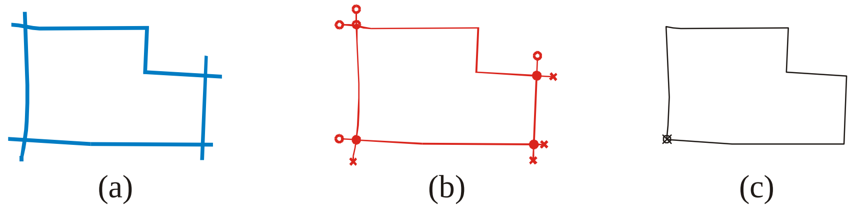


Figure 6.19 Segmentation of an object: (a) original object with four strokes, (b) segmented object with 11 segments, and (c) the object after the object completion process.

Each segment consists of a list of points, including a start and an end point. These boundary points are the only points that a segment can have in common with another segment. Accordingly, there are four types of segments: closed, self-closed, half-open, or open. To accelerate subsequent processing tasks, segments store pointers to their immediate neighbor segments.

6.3.2.3 Extraction of Symbolic Gestures

Besides gestures that initiate an action, such as the delete gesture (Section 6.1.4), people frequently use symbolic gestures to assign a specific semantics to sketched objects (Section 4.3.2). An example of a symbolic gesture is a triangulation point (e.g., ⟁) in a surveyor’s sketch. Typically, the set of symbolic gestures depends on the specific application domain; however, certain gestures are application-independent. The prototype understands two types of gestures: dashed lines and hatched areas (Figure 6.20).

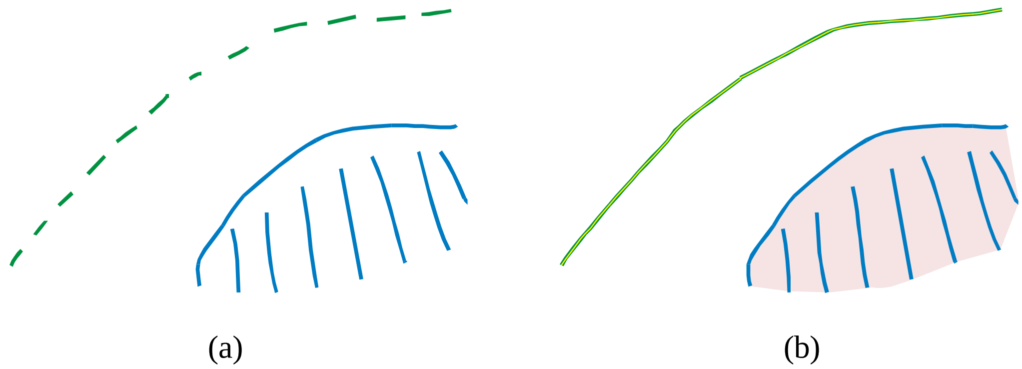


Figure 6.20 (a) Two symbolic gestures and (b) the prototype's interpretations.

6.3.2.4 Object Clean-Up and Completion

The prototype takes care of small inaccuracies before and during the segmentation of an object (Section 6.3.1); however, incompletely drawn objects of a certain magnitude are difficult to correct at that stage of the processing cycle, because strokes or segments are examined individually. The three most common problems are inadvertently open polygons (Figure 6.21a), interrupted lines (Figure 6.21b), and overshoots (i.e., short intersections) undershoots (i.e., gaps) (Figure 6.21c).

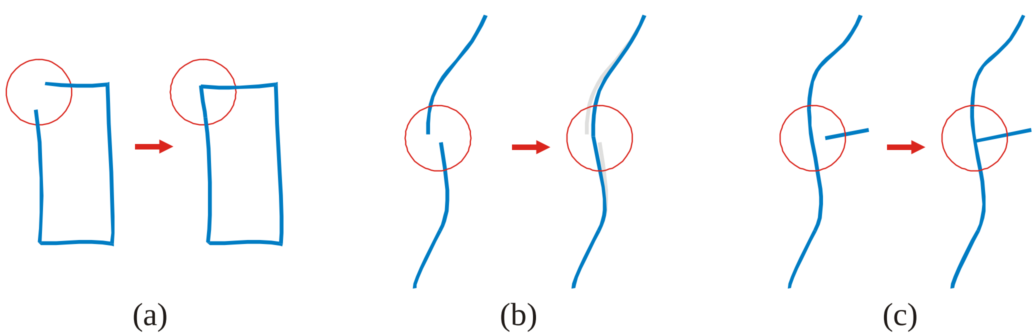


Figure 6.21 Three cases of object completion: (a) closing a polygon, (b) continuing an interrupted line object, and (c) connecting an undershoot to a base line.

The prototype detects such sketch deficiencies and corrects them automatically. After this clean-up procedure, the prototype attempts to connect segments in a meaningful way with each other.

6.3.2.5 Object Type Determination

Sketched objects are either of type region or line (Section 6.1.2). In an attempt to detect the type of an object, the prototype extracts all closed areas of a sketched object and stores them according to their size. The prototype rejects the hypothesis that an object is a region when no closed areas are found. This hypothesis is also rejected when the sum of the areas of all closed regions is small compared to the area of the convex hull of the object.

6.3.2.6 Centerline Extraction

Line objects are examined as to whether any line segments are approximately parallel to each other. If such segments are found, they are substituted by their centerline (Figure 6.22). The prototype stores centerlines and the remaining segments in a list of lines, sorted according to their length.

(a) (b)

Figure 6.22 Centerline extraction: (a) sketched object and (b) its interpretation with centerlines.

6.3.2.7 Kernel Extraction

The kernel of an object is a line-shaped abstraction of this object (Montanari 1968). If the object is a line, then the kernel is the result of the centerline extraction. For region objects, the prototype approximates the centerline of an object with the longer centerline of the object's *Tilted Minimum Bounding Rectangle* (TMBR). The TMBR is the bounding rectangle with the smallest possible area of the whole set of possible bounding rectangles for a specific object. The difference between a regular MBR and a TMBR is that the TMBR can have any orientation (α) in reference to the coordinate system (Figure 6.23), while the orientation of a regular MBR is fixed (i.e., parallel to the coordinate system).

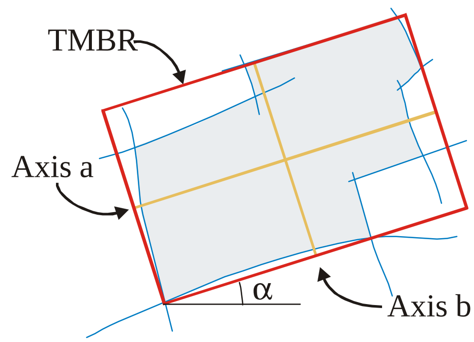


Figure 6.23 An object of type region and its TMBR with the two main axis a and b.

The calculation of the TMBR is slightly more complex than an ordinary MBR; however, a TMBR has *per se* the unique characteristic of fitting spatial objects optimally and the directional component provides a first evidence of a sketched object's orientation. If one takes into account that most objects in sketches have box-like shapes (Blaser 1998), then the TMBR is an ideal approximation for region objects in a sketch.

6.3.3 Assembling the Digital Sketch

The sketch parsing and object generating process is concluded when the user indicates that the sketch is complete and ready to be processed (Figure 6.24). By now the application has interpreted and simplified all sketched objects (Section 6.3.2). The next step is the generation of the association graph, which connects the sketched objects (Section 5.3.3) and the computation of the set of binary spatial relations (Section 5.1).

6.3.3.1 Association Graph Generation

The prototype generates the association graph according to the configured association graph model. The complete model includes every possible binary spatial relation between the sketched objects. Other models consider only subsets of this complete set (5.3.2).

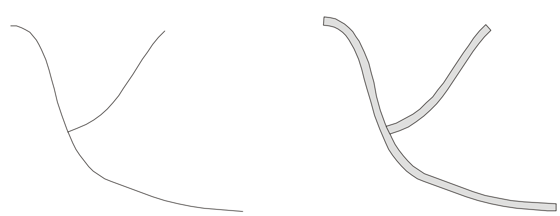
The system supports three different reduced association graph models. Within these models sketched objects that are connected:

- with their Voronoi neighbors
- with their temporal neighbors
- according to metric rules *(Blaser 1999a)*

The prototype allows a user to arbitrarily combine these reduced subsets to a combined association graph. The selected association graph is computed automatically and all necessary binary spatial relations are created without any user intervention.

6.3.3.2 Topology

The topology between two sketched objects is the first characteristic of a spatial relation to be assessed. Depending on the type of objects involved, there are different topological setups possible. The prototype considers *Region–Region* (RR) relations (Section 5.1). The topology for *Line–Line* (LL) relations and *Line–Region* (LR) are not implemented. Since line objects play an important role in spatial sketches (Blaser 1998), they are approximated as thin long regions (Figure 6.24). The prototype widens simple lines by a fixed amount; line objects with parallel segments are widened by the averaged width between the segments. This approach allows the prototype to treat all objects as regions and to compute the topology accordingly.



(a) (b)

Figure 6.24 (a) A line object and (b) its region-like representation.

Freehand sketches are rough approximations of spatial scenes in which sketched objects are frequently drawn inaccurately (Blaser 1998). The *actual* topology between two sketched objects may, therefore, differ from the *intended* topology. The prototype uses heuristics to correct this problem so that the result reflects a user's intention more accurately. This approach allows the application to determine the most likely topology for ambiguous cases. Figure 6.25 shows an example of two region objects that have a small intersection, leading to a topological *overlap* relations; however, it is more likely that the intended topology is of type *meet*, but the user did not draw the common boundary exactly.

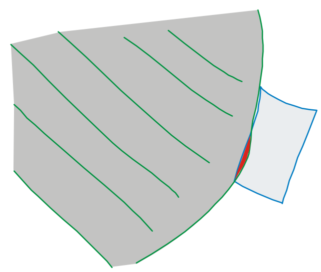


Figure 6.25 Topological *overlap* relation that is captured as a *meet* relation.

A correction of the topology extends only to the direct conceptual neighbors (Section 5.1.1) of a particular topological relation (e.g., *overlap* and *meet*, *contains* and *covers*, and *inside* and *coverdBy*) and the thresholds are dependent on a user's preferences.

6.3.3.3 Direction

The model for the direction relation between two objects is based on the spatial relation between the object's MBRs (Chapter 5.1.3).

6.3.3.4 Metric

The metric components of spatial relations are computed according to the formalisms discussed in Section 5.1.2. The specific set of metric components depends on the topology of a spatial relation. The prototype supports *Region–Region* relations; line objects are approximated as thin regions (Section 6.3.3.2) so that *Line–Region* and *Line–Line* relations can be processed using the formalisms for *Region–Region* relations

(Section 5.1.2). The computation of the metric components of the spatial relations is the last step in building the digital sketch.

6.4 Query Processing

The user can initiate the spatial query once the sketch is drawn and any optional editing in the object or diagram view is complete. During query processing the application compares all scenes in a selected dataset with the sketched query and computes for each sketch-scene pair a scene similarity value. The comparison is based on characteristics of sketched objects, their spatial relations, and the spatial scene (Egenhofer 1997). Each sketch is stored in an ASCII text file (Blaser 1999b), allowing external applications access to the data structure of the prototype. The query process involves three conceptual steps: *object association*, *scene similarity assessment*, and *result presentation* (Figure 6.26).

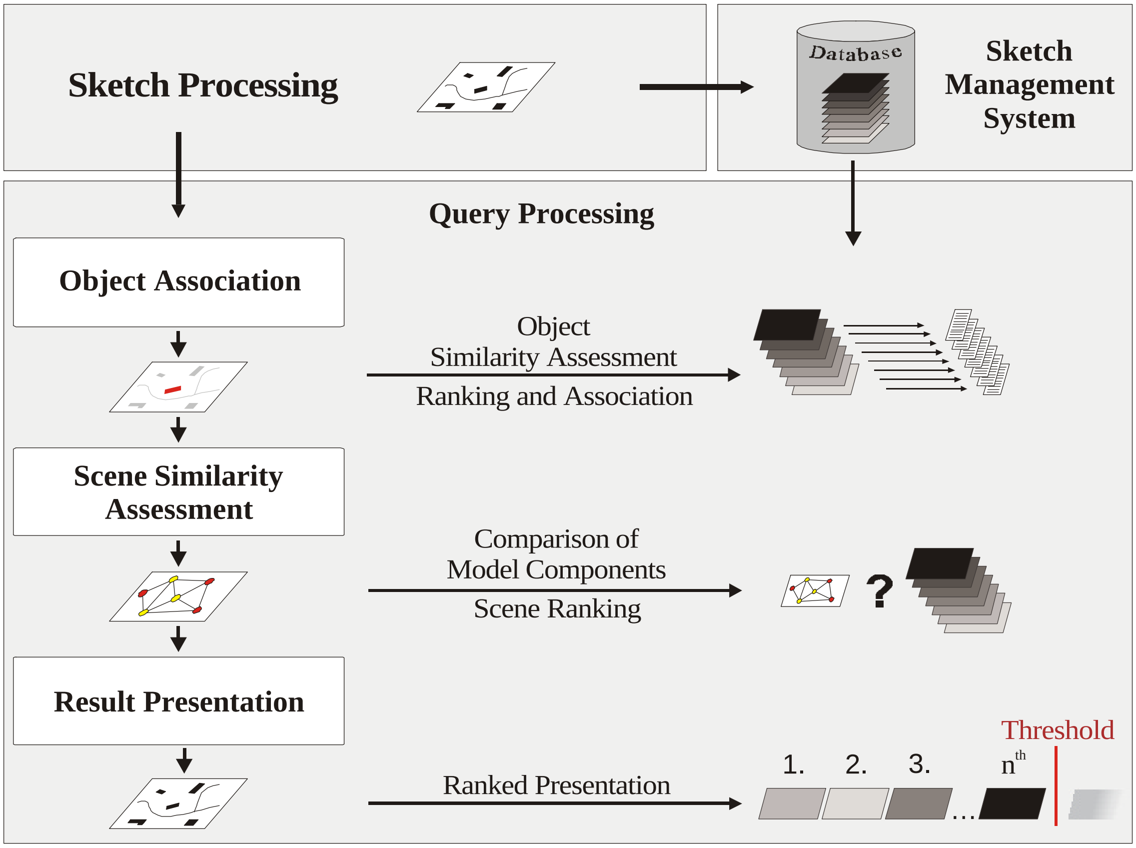


Figure 6.26 Query processing of the prototype.

6.4.1 *Object Association*

Object association is the process of linking each object in the sketched query with the most similar object in each sketch of the dataset. The result of this process is a *key-list* with binary object associations for each database record. This list is central for the subsequent assessment of scene similarities. Objects are associated according to their geometric similarity. The prototype takes into account the following object characteristics: the number of individual components, the ratio between the two axis of the TMBR and their orientation, the number of directional changes, the number of orthogonal angles, and the sum of the directional change. Additionally scale and dimension of an object are considered (i.e., the length for line and the area for region objects). The total similarity between two sketched objects (*geometric similarity*) is computed as the weighted average of individual geometric similarities.

6.4.2 Scene Similarity Assessment

The prototype computes the similarity of two spatial scenes (*scene similarity*) based on a comparison of their digital sketches (Section 5.5) at three different conceptual levels (Figure 6.27).

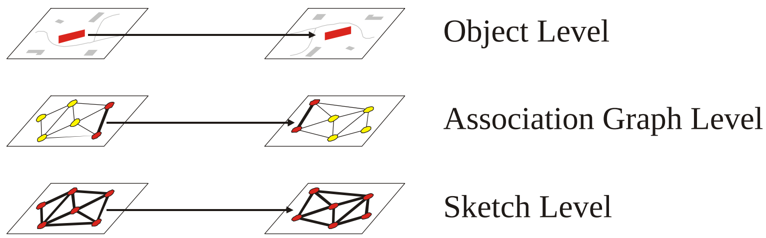


Figure 6.27 The three conceptual levels of the scene similarity assessment.

The comparison at the first level is based on the results of the object association process (Section 6.4.1), that is, on the geometric similarity of all associated object pairs. At the second level, the prototype assesses the similarity of corresponding spatial relations of the association graph. The comparison at the third level focuses on sketch specific properties. The key-list provides for all levels the framework for the similarity assessment.

6.4.3 Result Presentation

The result of the query is presented in the *result browser* dialog (Figure 6.28). The dialog includes a graphical and a text-based section. The left hand side of the dialog browser displays the sketched query and allows the user to modify parameters of the computational model. The right hand side is concerned with the query results. The graphical part of the dialog box permits the user to visually compare the sketched query with retrieved sketches from the database. The user can re-associate sketched objects and change their weights directly within the result dialog.

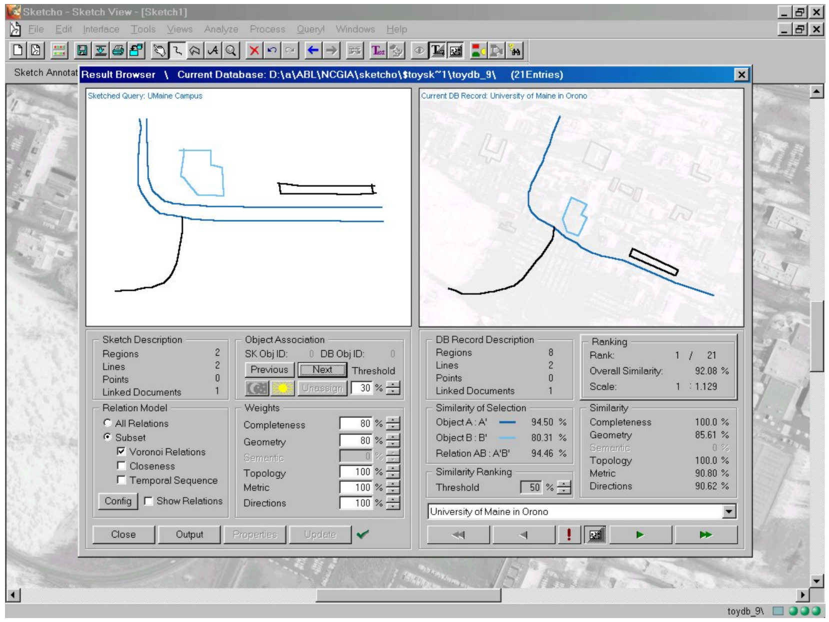


Figure 6.28 Result browser dialog of the prototype.

The prototype detects if the query or parameters of the computational model have changed and indicates that the query needs to be reprocessed. However, only those components are recomputed that are affected by the user’s changes. Besides comparing the results graphically in the result browser, the user can export the results as MS Excel conform text files or print them as a sorted list or graphics (Blaser 1999b).

6.5 Summary

This chapter discussed the prototype implementation of a sketch-based system to query spatial information in a GIS environment. The focus was on the implementation of the classes for the digital sketch, the sketch-based user interface, and on issues concerning the processing of a user’s sketched query. The prototype implementation provided evidence for the practicability and usefulness of the digital sketch and its supporting spatial theories. First experiences with the prototype implementation suggest also that a sketch-based user interface is a viable tool to specify and query spatial information.

Chapter 7
Model Evaluation

The reduced association graph of the digital sketch stores significantly less information than the complete association graph (Section 5.3.3). However, the hypothesis of this thesis asserts that the reduced amount of knowledge is sufficient to rank a set of spatial scenes according to their similarity to a sketched query. This chapter is to provide evidence for this hypothesis, applying an empirical approach.

The result of a comparison between a sketched query and a set of spatial scenes is a set of similarity values that can be used to sort the spatial scenes according to their similarity to the sketched query. This ranking list is an expression of how a specific model classifies a set of spatial scenes according to their similarities. Changing the model (e.g., by using a different association graph for the digital sketch) has an impact on the similarity values and the generated ranking list; therefore, this method can be used to compare different approaches to rank spatial scenes. In this context it is assumed that a similarity assessment based on the complete association graph produces an accurate and representative ranking list of sketched scenes, because it considers the set of *all* binary spatial relations.

There are basically two methods to compare two ranking lists: The first method focuses on comparing corresponding similarity values (i.e., the similarity values of each matching rank pair are compared), while the second approach is concerned with the rank differences only (i.e., the actual similarity values are irrelevant for this comparison). Within the scope of a sketched query, people are primarily interested in retrieving the most similar spatial scenes and they are less concerned with actual similarity values. Therefore, it is appropriate to consider the rank differences between corresponding spatial scenes for the evaluation of the reduced association graph.

7.1 Statistical Considerations

There exist several approaches to compute the correlation between ranking lists (Tate and Clelland 1957; Mosteller and Rourke 1973; Daniel 1978; Gibbons 1985; Kornbrot 1990). Most of these nonparametric statistical tests are designed to describe the correlation of different datasets in such fields as psychology, ecology, or material science. Conventional analyses in these application areas are concerned with a good correlation throughout the entire range of ranking lists; however, an evaluation of the correlation of ranking lists resulting from database queries (e.g., using a web search engine) is different. Here, the primary focus is on the first few ranks, because the relevance of retrieved items decreases rapidly for lower ranks. The number of ranks that are important is independent of the number of totally retrieved data items in so far that only good matches are relevant–dependent on the query and on the content of the database, this number may, therefore, vary considerably.

The fact that not all positions of a ranking list have the same relevance makes it difficult to prove that the first portion of two such lists correlate if conventional methods are used and if the lists start to diverge after a certain point. To overcome this problem we introduce additional analysis methods that consider only a specific section of the ranking lists. Therefore, the statistical analysis of the correlation between the scene similarity computed with the complete and the reduced association graph is based on two well-known nonparametric tests (Sections 7.1.1 and 7.1.2) *and* an additional evaluation that relies on generic statistical variables (Sections 7.1.3). The rank differences that are used for the evaluation are obtained by subtracting the rank resulting from the reduced association graph from the rank of the complete association graph (i.e., `Difference`

`= CompleteRank - VoronoiRank`). However, the sign (+/-) of the rank difference is only relevant for the assessment of the adjusted average value (Sections 7.1.3).

7.1.1 Spearman Rank Correlation Test

The *Spearman Rank Correlation Test* compares two entire ranking lists (Tate and Clelland 1957). The resulting Spearman Rank Correlation Coefficient is a parameter that can take a value between –1 and +1, where +1 signifies perfect agreement between the two samples, while –1 signals complete disagreement (i.e., the two ranking lists are inverse). A value of 0 means that there is no association between the two samples. The computation of the Spearman Rank Correlation Coefficient (*R*) (Equation 7.1) and the significance value (Equation 7.2) are straightforward.

$$R = 1 - \frac{6\sum_{i=1}^{n}(U_i - V_i)^2}{n(n^2 - 1)} \tag{7.1}$$

$$z = R \cdot \sqrt{n-1} \tag{7.2}$$

with: R = Spearman Rank Correlation Coefficient

U_i, V_i = Paired values (corresponding ranks)

n = Number of ranks

z = Approximation of the standardized normal variable

A good correlation between the ranking lists of the reduced association graph and the complete association graph is characterized by a positive Spearman Rank Correlation Coefficient close to 1. The result (z) is an approximation of the standardized normal variable that can be compared with a table for the standard normal distribution. The value indicates the probability that the two ranking lists correlate.

7.1.2 *Wilcoxon Signed-Rank Test*

A nonparametric test that was especially developed to assess the correlation between two ranking lists is the *Wilcoxon Signed-Rank Test* (Daniel 1978; Gibbons 1985). This statistical test is designed to compare the location of the median of two populations of samples. The test often involves the use of matched pairs, such as the corresponding ranks in two ranking lists, for which it tests for a median difference of zero. The Wilcoxon Signed Ranks Test does not require that the population is normally distributed; however, the test assumes that the population distribution is symmetric.

The Wilcoxon Signed Ranks Test is more sensitive than the Spearman Rank Correlation Test; however, it is also more complex to compute. There are tables for small populations up to 30 samples. Equation 7.3 has to be used to obtain an approximation value for the significance of the test if the number of pairs exceeds this limit (Daniel 1978).

$$z = \frac{T - 0.5 - n(n+1)/4}{\sqrt{n(n+1)(2n+1)/24}} \tag{7.3}$$

with n = Number of different ranks

T = Sum of Wilcoxon ranks

z = Approximation of the standardized normal variable

The value T in Equation 7.3 is computed based on the sum of all positive and all negative *Wilcoxon Ranks* (Daniel 1978). The larger absolute value of both sums is T. The result (z) is an approximation of the standardized normal variable that can be compared with a table for the standard normal distribution. A high value of z indicates that the two ranking lists have *no* correlation, which is the opposite of the standardized normal variable z produced by Equation (7.2).

7.1.3 Adjusted Average and Standard Deviation

The Spearman Rank Correlation Test and the Wilcoxon Signed-Rank Test are designed for evaluations of entire ranking lists, but they are inadequate to test the correlation of a particular subsection of a ranking list. This observation is relevant, because for the comparison of the reduced association graph with the complete graph only the first portion of the ranking list is relevant (Section 7.1). A method that allows an evaluation of a specific subsection of two ranking lists is to compute the *average* and *standard deviation* of only those ranked pairs that fall within a certain range (e.g., ranks 1 to 10). To evaluate the obtained parameters with respect to the size of the ranking lists they can be normalized using the size of the ranking list (Equations 7.4 and 7.5). The parameter *m* represents the size of the subsection, with $m > 1$ and $m \leq n$. For $m = n$ the average value (μ_m) is 0, while σ_m is equal to the standard deviation of the entire set of rank differences.

$$\mu_m = \frac{1}{n} \frac{\sum_{i=1}^{m} (U_i - V_i)}{m} \tag{7.4}$$

$$\sigma_m = \frac{1}{n} \frac{\sum_{i=1}^{m} (U_i - V_i)^2}{m-1} \tag{7.5}$$

with U_i, V_i = Paired values (case corresponding ranks)

m = Number of matched pairs up to the threshold

n = Number of matched pairs in the ranking lists

μ_m = Adjusted average up to the threshold

σ_m = Adjusted standard deviation up to the threshold

This approach to consider only a part of the rank differences allows a qualitative statement regarding the correlation of the *relevant section* of the ranking lists while taking into account the size of the entire ranking list.

7.2 Setup

The test bed for the evaluation of the reduced association graph is the prototype implementation of the sketch-based query processor (Chapter 6). The base for the evaluation is a set of five sketch datasets. Each dataset consists of a number of sketches (data records), containing the same set of objects; however, the location of these objects was modified so that all sketches are different (Blaser 1999b). The individual scene similarities are obtained by comparing each sketch in a dataset with a sketched query (Section 6.4.2). The weights for all components of the digital sketch are equal (100%). To obtain accurate results for the evaluation of the association graphs, only components involving spatial relations are used (i.e., topological, metric, and direction relation).

The result of each query is a ranking list with the sketches in the dataset sorted according to their similarity to the sketched query. The most similar sketches are on top of the list. This procedure is repeated for the complete and the reduced association graph. Considering the ranking list that results from the query with the complete association graph as a base (i.e., assuming a correct ordering for this list), the rank differences are computed by subtracting the associated ranks of both lists from each other (Equation 7.1). These lists of rank differences (one for each dataset) are subsequently evaluated according to the three statistical methods described in Section 7.1. A relative and an absolute threshold for the adjusted average and the adjusted standard deviation was introduced. The absolute threshold is at 7 ranks (10% of the smaller datasets) and the relative threshold is at a position representing 10% of the number of records of the individual datasets. These thresholds are set such that they represent a typically relevant portion of a ranking list.

Table 7.1 provides an overview of the five datasets, considering their contents, and the methods and criteria used for their creation.

Dataset	Number of Records	Objects per Record	Creation Type	Constraints	Variation	Object Density
1	70	7 regions 3 lines	manual	geo-spatial	low	medium
2	70	7 regions 3 lines	manual	geo-spatial	medium	medium
3	70	7 regions 3 lines	random	no constraints	high	medium
4	1000	6 regions	random	no constraints	high	low
5	1000	24 regions	random	no constraints	high	high

Table 7.1 Composition of the five test datasets, and description of the methods and criteria used for their creation.

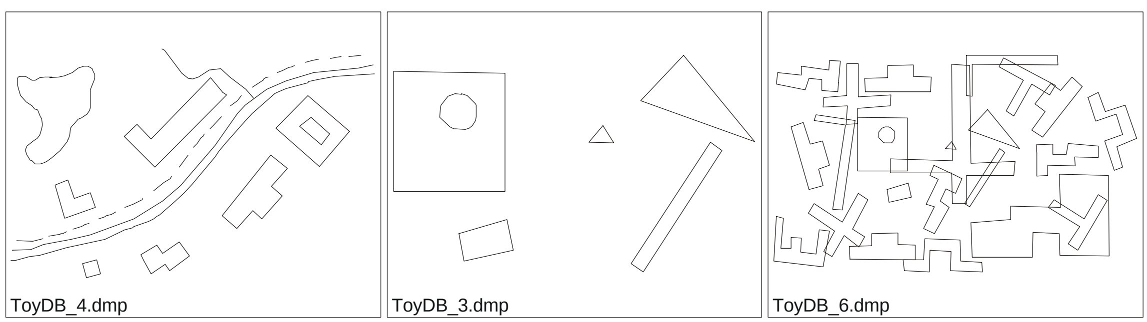
The first dataset (*geo-low-variation dataset*) consists of 70 sketches with sketched objects and spatial configurations that are common in geo-spatial sketches (Blaser 1998) (Figure 7.1a). Each individual sketch in the dataset was created by modifying the location and orientation of the sketched objects manually and according to realistic criteria (e.g., sketched houses do not overlap). Individual modifications are small so that the variation between sketches in this test dataset is low.

The second dataset (*geo-medium-variation dataset*) was created using the same initial setting and the same set of sketched geo-spatial objects as for the *geo-low-variation* dataset (Figure 7.1a); however, the individual modifications between the sketches are greater, resulting in a higher overall variation for the initial spatial situation. Therefore, there are fewer sketches similar to the initial sketch than in the first dataset. This dataset contains also 10 sketches that are topologically and metrically identical to the sketched query but rotated by different angles.

The third dataset (*geo-random-high-variation dataset*) contains the same set of objects as the *geo-low-variation* and the *geo-medium-variation* datasets;; however, these sketches were created by applying random changes (e.g., move and rotate) to all objects. The result is a dataset with a high variation between the spatial configurations of individual scenes and sketches that are unconstrained considering their spatial configuration (e.g., streets may intersect with buildings).

The fourth dataset (*large-random-low-density dataset*) consists of 1000 sketches, each constructed from a small number (6) of regular geometric figures (Figure 7.1b), such as circles, squares, or rectangles. All sketches in this dataset were randomly generated so that there are no geo-spatial constraints and the variation between individual sketches is high. However, the density of the spatial scenes is low, because there are only a few geometric figures involved.

The fifth dataset (*large-random-high-density dataset*) consists of 1000 sketches with regular geometric figures as well. Besides the set of objects used for the *large-random-low-density* dataset, there are 18 additional objects in each sketch (Figure 7.1c). The higher number of objects increases the density in the data records and results in a larger number of non-disjoint relations between objects. The goal of this approach is to create some “noise” and to obstruct objects from each other. While this has no influence on the similarity assessment using the complete association graph, it affects the assessment based on the reduced graph, because not necessarily all relations are considered (Section 5.4.1).



(a) (b) (c)

Figure 7.1 The three initial sketches for the five test datasets.

7.3 Results

Tables 7.2 and 7.3 summarize the results of the comparison of the complete and the reduced approach to assess the similarity between spatial scenes, for each of the five datasets. Table 7.2 depicts the absolute results and Table 7.3 the normalized values. The normalization is based on the size of the individual ranking lists. Both tables consist of three categories. The first category considers the first seven ranks (absolute), the second category considers the first 10% of all ranks in a dataset (relative), and the last category considers the entire ranking list. The adjusted average is substituted in the third category by the absolute maximum average, because the adjusted average is of course zero if entire ranking lists are considered. The maximum average reflects the highest deviation from zero, observed at any point between the first and the last corresponding rank.

	First 7 Ranks		First 10%		Entire Dataset			
Dataset	*Average*	*StdDev*	*Average*	*StdDev*	*MaxAvg*	*StdDev*	*Spearman*	*Wilcoxon*
1	-0.14	1.57	-0.14	1.57	-1.67	5.65	0.971	0.065
2	0.00	1.29	0.00	1.29	-4.47	12.33	0.816	0.144
3	-2.57	3.15	-2.57	3.15	-5.00	14.13	0.759	0.248
4	-0.86	1.46	-61.69	83.14	-93.26	195.59	0.771	1.114
5	-12.14	14.32	-51.85	69.41	-76.59	205.97	0.746	1.976
Average	**-3.14**	**4.36**	**-23.25**	**31.71**	**-36.20**	**86.73**	**0.813**	**0.709**

Table 7.2 Summary of the statistical evaluation of the rank differences for each of the four test datasets.

	First 7 Ranks		First 10%		Entire Dataset			
							***P*-Value**	
Dataset	*Average*	*StdDev*	*Average*	*StdDev*	*MaxAvg*	*StdDev*	*Spearman*	*Wilcoxon*
1	-0.2%	2.2%	-0.2%	2.2%	-2.4%	8.1%	100.0%	94.8%
2	0.0%	1.8%	0.0%	1.8%	-6.4%	17.6%	100.0%	88.4%
3	-3.7%	4.5%	-3.7%	4.5%	-7.1%	20.2%	100.0%	80.3%
4	-0.1%	0.1%	-6.2%	8.3%	-9.3%	19.6%	100.0%	26.8%
5	-1.2%	1.4%	-5.2%	6.9%	-7.7%	20.6%	100.0%	5.0%
Average	**-1.0%**	**2.0%**	**-3.0%**	**4.8%**	**-6.6%**	**17.2%**	**100.0%**	**59.1%**

Table 7.3 Summary of the normalized statistical parameters of all four test-datasets.

Figures 7.2 and 7.3 show the distributions of the rank and similarity differences, and the rank correlation for the five datasets.

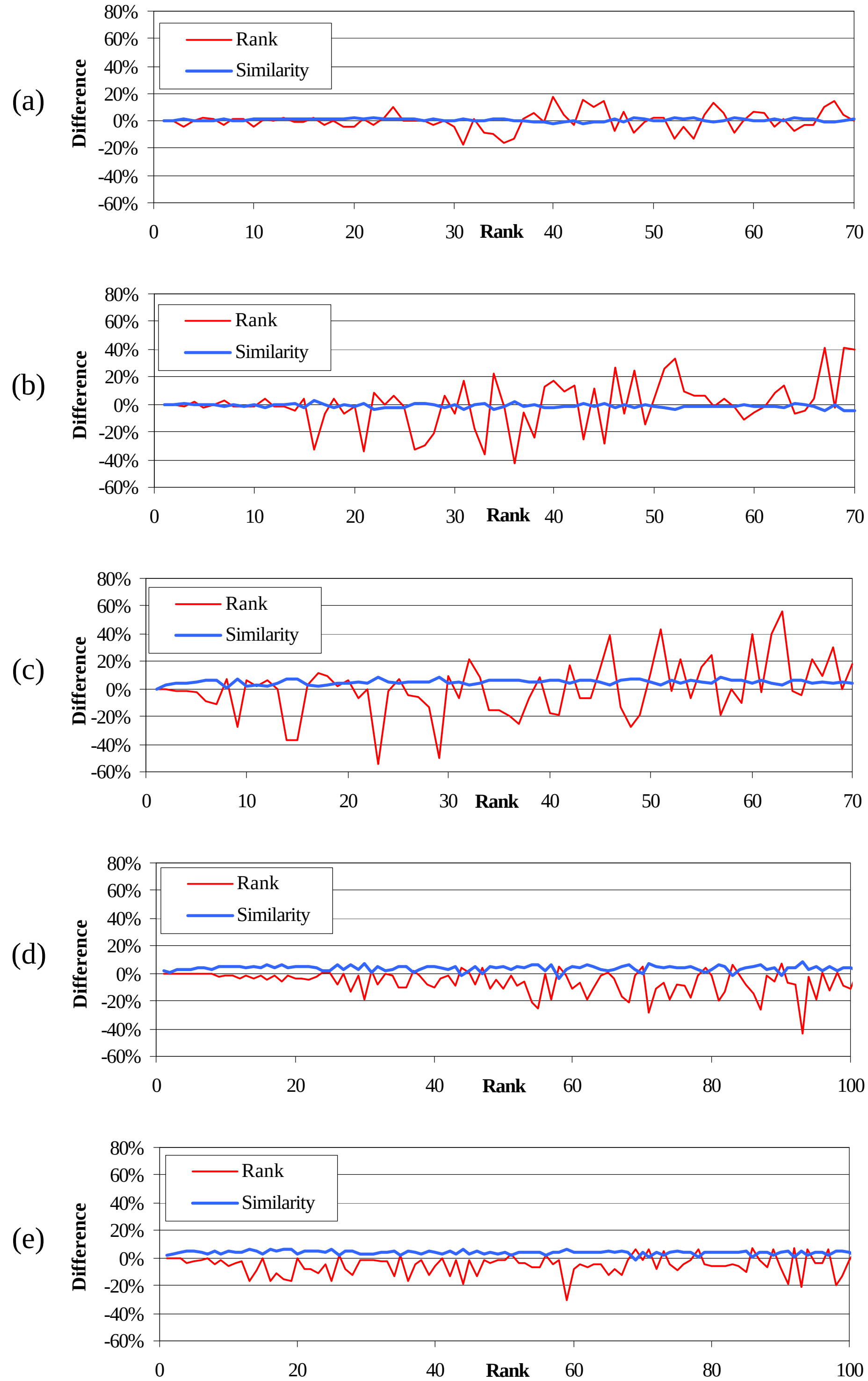


Figure 7.2 Rank and similarity differences for (a) *geo-low-variation*, (b) *geo-medium-variation*, (c) *geo-random-high-variation*, (d) *large-random-low-density*, and (e) *large-random-high-density* dataset.

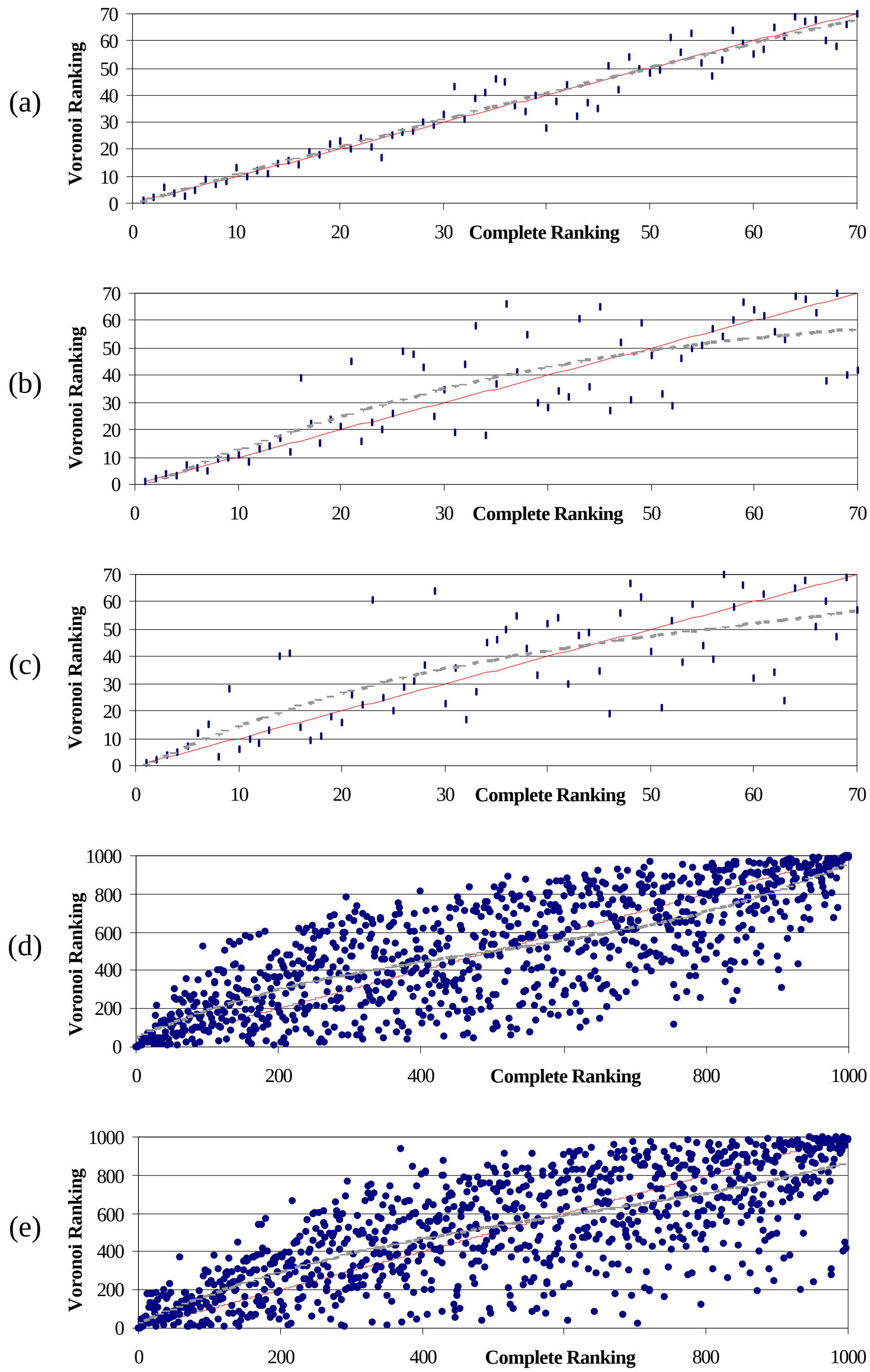


Figure 7.3 Rank correlation and trend line for (a) *geo-low-variation*, (b) *geo-medium-variation*, (c) *geo-random-high-variation*, (d) *large-random-low-density*, and (e) *large-random-high-density* dataset.

Table 7.4 shows the normalized results of individual components of the digital sketch, including topology, metric, and directions as well as scene similarity.

Dataset	Entire Dataset							
	Scene Similarity		**Topology**		**Metric**		**Directions**	
	Average	*StdDev*	*Average*	*StdDev*	*Average*	*StdDev*	*Average*	*StdDev*
1	0.60 %	1.02 %	0.90 %	0.93 %	8.39 %	5.62 %	10.11 %	6.30 %
2	-1.14 %	1.53 %	1.32 %	1.64 %	1.89 %	1.69 %	-6.62 %	3.83 %
3	4.89 %	1.69 %	4.92 %	2.19 %	7.85 %	2.32 %	2.25 %	2.44 %
4	2.81 %	2.90 %	5.31 %	3.56 %	3.49 %	3.04 %	-0.37 %	5.84 %
5	3.01 %	1.98 %	4.31 %	2.00 %	3.86 %	1.89 %	0.87 %	4.69 %
Average	**2.03 %**	**1.82 %**	**3.35 %**	**2.06 %**	**5.09 %**	**2.91 %**	**1.25 %**	**4.62 %**

Table 7.4 Normalized values for the topology, metric, and direction component differences as well as for the differences in scene similarity.

7.4 Interpretation

The following two sections interpret the results from a qualitative and quantitative point of view. For a qualitative assessment we consider the rank differences, whereas the quantitative assessment is based on the actual values of the similarity assessment that lead to the different ranking lists.

7.4.1 Qualitative Considerations

The subsequent observations and their discussion are based on an evaluation of the results obtained by comparing the order of corresponding ranks in the two ranking lists. The results are depict in tabular form in Tables 7.2 and 7.3, and graphically in Figures 7.2 and 7.3.

Observation 1: *The two rankings have a significant correlation.*

The Spearman Correlation Coefficient suggests for all datasets an acceptance of the hypothesis that the two ranking lists correlate (Table 7.3). This is also evident by considering Figure 7.3. The Standard Deviations over the entire datasets provide another indication for a correlation: The average Standard Deviation is 19.5%. This value is considerably below the corresponding value for a random distribution, which is around 40% (empirically evaluated).

Observation 2: *The first few positions of the two ranking lists show an excellent correspondence.*

Regardless of the size and the grade of variation of a dataset, the most similar spatial scenes (e.g., the first seven ranks in Table 7.3) are classified in close proximity to each other.

Observation 3: *Datasets containing spatial scenes that are similar to the sketched query result in a high correspondence of the ranking lists.*

Datasets with a low to medium variation (Figures 7.3a and 7.3b) have a better correlation than those with a high variation (Figures 7.3c, 7.3d, and 7.3e). This observation is also evident from the adjusted average and the adjusted Standard Deviation for the first portion of the ranking lists (Tables 7.2 and 7.3), which represents the most similar spatial scenes and which is significantly below the values for the entire datasets. The Wilcoxon Signed Rank Test provides support for this statement as well. This test indicates a likelihood of 94.8% (*geo-low-variation dataset*) and a likelihood of 88.4% (*geo-medium-variation dataset*) that the rank differences of these datasets have a median of zero. The difference between low and high variation datasets is also evident by considering the Standard Deviation for the entire datasets.

Observation 4: *The complete and the reduced approach sort the spatial scenes in the datasets into similar and dissimilar scenes. Spatial scenes that are similar to the sketched query are classified similarly, while those that are dissimilar are classified differently.*

This statement is based on the distribution of the rank differences around the hypothetical value of zero. Considering either Figure 7.2 or Figure 7.3 it is evident that rank differences are consistently lower in the first section (most similar) of the rank differences list

Corollary 1: *The relevant portion of two ranking lists corresponds well.*

This statement follows from Observation 3 and Observation 4, because the relevant portion of the ranking list is that section of the list that contains the most similar spatial scenes. Dissimilar scenes are of subordinate interest, which follows from the original purpose of a query.

Observation 5: *The two ranking methods agree for the most similar and the most dissimilar spatial scenes.*

The first part of this statement follows from Observation 4, while the second part can be deduced from Figure 7.2 and 7.3. The agreement on the most dissimilar spatial scenes is less significant than that of the agreement on the most similar spatial scenes. Figure 7.4 describes a generic distribution of the rank differences.

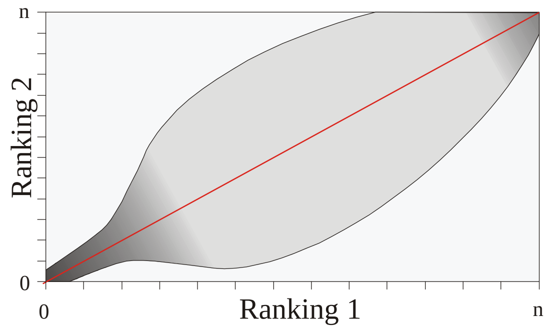


Figure 7.4 Typical distribution of the rank differences.

Dark shaded areas in Figure 7.4 indicate a high agreement between the two different ranking methods. The actual shape of the rank distribution depends on various factors, including the number of similar and dissimilar scenes and the grade of variation of the spatial scenes from the sketched query.

Observation 6: *The approach using the reduced association graph assesses rotated scenes with a consistently higher similarity value than the complete approach.*

The second dataset (*geo-medium-variation dataset*) contains 10 spatial scenes that are rotated but otherwise topologically and metrically identical to the sketched query (Section 7.2). The rotated scenes were considered more similar to the sketched query by the reduced approach, with an average of 11.5 ranks (16%). The reason for this phenomenon is that the reduced approach is only concerned with the immediate neighbor objects, which results in a higher direction similarity than when all objects are taken into account. Another factor is the chosen method for the direction similarity assessment, which is based on MBRs and which is not accurate when an object's MBR is inside a neighboring object's MBR.

Observation 7: *The reduced approach produces typically lower scene similarity values than the complete approach.*

This statement is based on Table 7.3. The methods to assess the metric and topological similarity produce more distinct values when only neighborhood relations are considered, which translates in higher rank differences (i.e., the similarity value for the metric and topology components of the reduced approach is smaller). The direction relation, however, produces a different result. Depending on the variation between the sketched query and individual scenes in the dataset the result may vary, which is indicated by the non-consistent value for the average of the direction value (-6.62% to +10.11% in Table 7.3). Another reasons for this variation is the direction assessment method (Section 5.1.3) that is based on MBRs.

Observation 8: *The trend for the rank differences within the first part of the ranking lists is negative.*

This statement follows from an analysis of the trend lines of the rank correlation graphs in Figure 7.3. The rational for this phenomenon is based on Observation 7, because the reduced approach classifies certain spatial configuration lower than the complete approach, which results in a distortion of the trend line

Observations 2, 3, and 4, and Corollary 1 provide evidence that the reduced set of spatial relations captures the relevant portion of a sketch sufficiently and that this subset of relations is an appropriate base for the association graph of the digital sketch. The summary result of this evaluation is:

The ranking lists for a spatial query with the reduced association graph and the complete graph are similar, particularly over the first few ranks, where the similarity between the sketched query and the ranked datasets is high. Thereafter the two lists do still correlate, but the rank differences are not significantly distributed around the hypothetical median value of zero.

The significant portion of the ranking list is that part of the list where the similarity between the sketched query and the sketches in the database is high. Because our findings indicate that there is a high correlation in this first significant portion of the ranking lists, we can *accept the hypothesis* formulated in Chapter 1. Hence, we also have found evidence that there is at least one subset of binary spatial relations that can substitute the complete set of binary spatial relations in the digital sketch (i.e., should the digital sketch be used to assess the similarity between a sketched query and sketches in a database).

7.4.2 Quantitative Considerations

The individual components of the digital sketch are the base for the assessment of the similarity between the sketched query and a set of sketches (Section 5.2.1). Because the individual similarity values for each component are known, one can compare their values for the complete and the reduced set of relations. Table 7.3 in Section 7.3 summarized the normalized differences of the topology, metric, and direction component, as well as the scene similarities for all five datasets. Figures 7.5 and 7.6 depict the relationship between the individual values graphically.

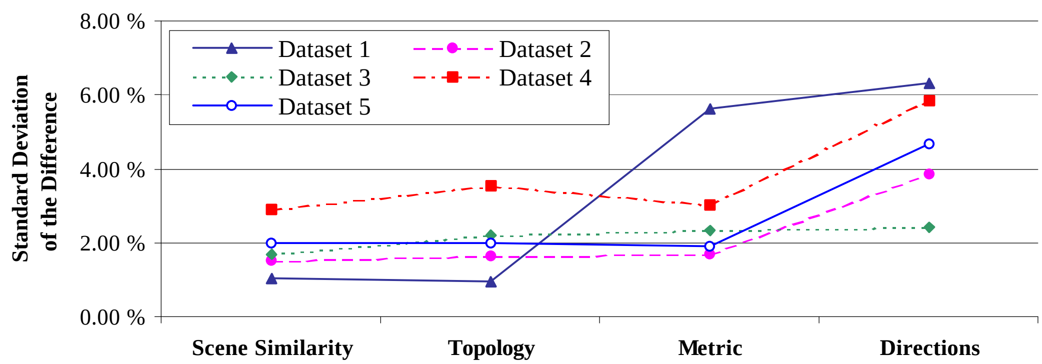


Figure 7.5 Standard deviation of the differences of individual model components.

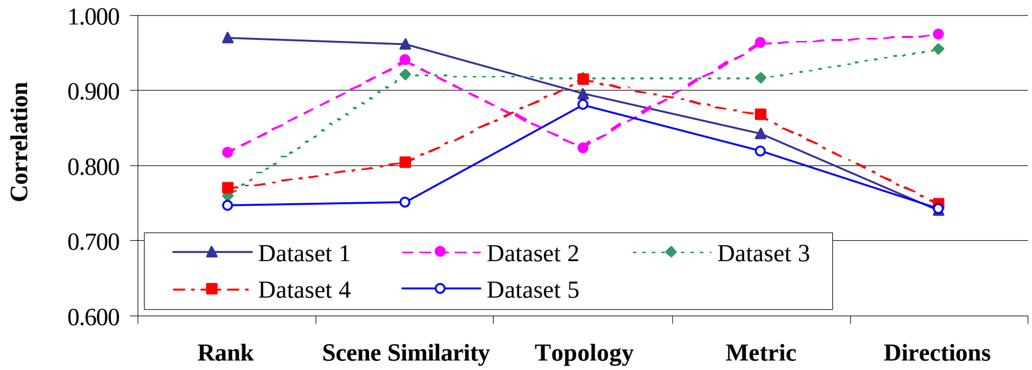


Figure 7.6 Correlation of the individual model components compared with the correlation of the ranks and the similarity values.

The values of the topology, metric, and direction components are interconnected, because these components are combined to obtain the scene similarity between two sketches (Section 6.4.2). The scene similarity, on the other hand, is the base of the ranking list, which is then used to compare the individual model components (of the two different assessment approaches).

In general, a high standard deviation indicates that a component is assessed differently, whereas a low standard deviation is a sign that both methods have interpreted the component similarly. The correlation, on the other hand, is an indication how consistently the two approaches have assessed a component. For instance, dataset one (*geo-low-variation*) has a high standard deviation for the metric component and a low correlation, indicating that the metric component was assessed inconsistently by the two approaches. Dataset four (*large-random-low-density*), on the other hand, has a high correlation and a high standard deviation for the topology component, which indicates that the two assessment approaches have consistently produced different values for this model component.

Another factor that has an influence on the value of individual model components is the method of creation and the grade of variation of the spatial scenes in relation to the sketched query. The first dataset (*geo-low-variation*), for instance, contains many scenes where only few objects were moved in relation to the sketched query. This results in a high metric and direction similarity for the complete approach, where the large number of unaffected spatial relations dampens the result, but a moderate similarity for the reduced approach, where individual relations have more weight.

The direction component has always a higher standard deviation than the other components. The rationale for this finding is that the direction component leads always to a meaningful description of a relation, because there is no limiting, distance-dependant factor. This characteristic has a positive effect for the approach, using the complete association graph, because the description of the spatial scene becomes more constraint and, therefore, more stable; however, for a sketched scene this might also be a disadvantage, because direction relations between faraway objects are often vague and unintended so that such a scene may become over-constrained.

7.5 Summary

This chapter evaluated the reduced association graph for its suitability to capture the spatial configuration of spatial scenes in the context of sketch-based queries. For this purpose the reduced association graph was compared with the complete graph. This evaluation was based on a relative comparison of ranking lists that were obtained by querying datasets containing sketches and using both approaches to order the sketches according to their similarity with sketched queries. A total of five different sketch datasets were used. The statistical analysis was based on the Spearman Rank Correlation Test, the Wilcoxon Signed-Rank Test, and on the adjusted average and adjusted standard deviation. The result of this analysis showed that the first portion of the ranking lists is similar for both approaches. Because the ranking lists order the sketches according to their similarity to the sketched query, with the most similar scene first, the first portion of the ranking list is also the most significant portion. This implies that the first few positions of a ranking list are most relevant, confirming the hypothesis that the reduced association graph is an appropriate substitute for the complete graph in the context of the digital sketch.

Chapter 8

Conclusions

This chapter reviews objectives, methodology, and results of this thesis, and discusses possible future research. Section 8.1 is structured according to the four main topics of this thesis and provides an overview of the research of this thesis. Section 8.2 highlights the major results of this thesis and reasons about implications of this research. Section 8.3 discusses possible future research topics and lists questions that have been raised through this research. The thesis closes with a conceptual description of an integration of different multimedia data types, inclusive sketches (Section 8.4).

8.1 Summary

The global objective of this thesis was to create the theoretical foundation for a sketch-based system to query spatial information in a GIS. In this context there were four main areas to investigate: (1) the interplay of different user modalities in a GIS, (2) people's sketching habits, (3) methods to translate a sketched scene into a digital representation, and (4) the practicability of the Spatial-Query-by-Sketch concept. The following four sections summarize the investigations in each of the four areas.

8.1.1 Human-Computer Interaction in GIS

The analysis of human-computer interaction in GIS focused on user modalities, user actions, and common user operations. The objective of this investigation was to evaluate the potential of traditional and alternative user modalities. In an initial approach we investigated advantages and limitations of widely used modalities, such as typing and pointing, and alternative modalities, such as talking and sketching. This examination was based on observations of how people interact with each other and with computing devices, as well as on findings from previous research. The second part focused on issues concerning the suitability of user modalities for specific user actions. For this purpose, complex user interaction tasks were broken into elementary user actions so that they were easier to analyze and classify. The result of these examinations was a set of guidelines for the design of user interfaces in GIS that identify appropriate user modalities for specific user actions.

8.1.2 People's Sketching Behavior

To study the sketching behavior of people, we surveyed 37 human subjects with different origin, gender, age, and profession. The goal of this survey was to gather evidence about how people sketch, what techniques they use, and how a typical sketch is composed. Each human subject was asked to draw three sketches based on a written description. The thematic of these three written scenarios was chosen such that it is possible to cover a wide variety of aspects of spatial sketches. Each sketch was analyzed on an object-by-object base. Besides sketched objects we investigated also their binary spatial relations and sketched annotations.

8.1.3 Concept of the Digital Sketch

The digital sketch is a model of a sketched, spatial representation that captures geometric attributes of sketched objects and properties of binary spatial relations between them. The objective was to formalize an efficient model that captures all relevant characteristics of a sketched scene and that can be used as a base for comparing a sketched scene with a set of spatial scenes in a database. The model is founded on findings from the survey (Chapter 4), spatial formalisms from previous research (Section 5.1), and on new formalisms developed in this thesis. The model takes into account the geometry of sketched objects and their binary spatial relations, notably their topological, metric, and direction relations. To increase the efficiency of the model, a method was developed that allows a discrimination of relevant binary spatial relations. The result of this evaluation of spatial relations is a reduced association graph, consisting of all sketched objects as nodes and a subset of the complete set of binary spatial relations as edges. The method that was used for the selection of the subset of relevant spatial relations is based on the spatial neighborhood of sketched objects and on a constrained Delaunay Triangulation.

8.1.4 A Sketch-Based Query System

The prototype application of the sketch-based query system is the implementation of ideas, formalisms, and concepts that were investigated in this thesis. The objective of the implementation was to create a platform that would prove the usability of the concept of Spatial-Query-by-Sketch and that would allow gathering experimental evidence for testing the hypothesis of this thesis. The prototype implementation was written in C++, using Microsoft's MFC libraries. The application consists of a sketch-based user interface, a sketch analysis engine, a query processing module, and a query result browser. Multiple views of the sketch allow a user to interact with the sketched query at different levels of abstraction and to follow the automatic interpretation of the sketch by the system.

To evaluate the hypothesis of this thesis a set of sketched queries was processed against five datasets with 70 to 1000 data records (i.e., individual sketches). Each dataset was queried twice, once using the reduced association graph and once using the complete association graph as a base for the digital sketch. The sketched queries included 6 to 10 line and region objects and the queried datasets between 6 and 24 objects.

8.2 Major Results

Research that was done in the scope of creating the theoretical foundation for a sketch-based query system in GIS led to a variety of findings that can be summarized as six individual result statements.

- *The reduced association graph represents a relevant subset of all spatial relations in a sketch.*

  An association graph is an abstract network representing sketched objects and binary spatial relations between them. The reduced association graph is obtained by considering only those binary spatial relations in a sketch that connect objects that are spatial neighbors. Spatial neighbors share at least on edge in a constrained Delaunay Triangulation (with the objects' outlines as the constrained edges). An experimental comparison between the complete and the reduced association graph showed that a sketched query based on the complete association graph and a sketched query based on the reduced association graph produce a similar ranking of the sketches in the database. This leads to the conclusion that the reduced association graph represents the relevant subset of all binary spatial relations between objects in a sketched scene.

  We have found a high rank correlation between corresponding spatial scenes in the ranking lists where the similarity between the sketched query and the spatial scenes in the database was high. This indicates that the *most relevant* portions of the ranking list are similar. This portion of the ranking list is relevant, because the primary targets of a typical database query are the *most similar* datasets of a database. This result also supports the hypothesis of this thesis, which asserts that there is at least one subset of binary spatial relations that, when used to query a database, leads to a similar result as a query that relies on the complete set of spatial relations.

  This result is significant, because it suggests that only a subset of all binary spatial relations is necessary and sufficient to describe a spatial scene appropriately. Should a system want to compare different spatial scenes, it can focus on those spatial relations that are relevant, while disregarding those that are less important. This approach leads to an *increase of the efficiency of the system* and presumably also to better results, because only relevant information is considered.

- *The digital sketch is a concise representation of a spatial sketch.*

  The digital sketch captures the geometry of objects and the spatial configuration of a sketch. The reduced association graph is the framework of the digital sketch in that it indicates which objects (nodes) are interconnected through binary spatial relations (edges). The graph binds all sketched objects to the connected network structure (i.e., there are no disconnected objects). The network of sketched objects and spatial relations is intuitive, because objects are connected only with their spatial neighbors. We have shown that the number of components of the reduced association graph increases only linear by O(n) when sketched objects are added. The size of the complete association graph, on the other hand, increases by $O(n^2)$. As a consequence, a digital sketch that is based on the reduced association graph is more efficient for storing and processing than a digital sketch that relies on the complete graph. This result is significant for systems dealing with spatial scenes that contain objects and their spatial relations. The significance increases for larger systems and the findings suggest that it is possible to store binary spatial relations in spatial databases.

- *Geo-spatial sketches consist typically of a small number of simple and abstract geometric figures.*

  Our survey showed that people typically use a small number of sketched objects—between 12 and 17 objects—to describe a spatial situation. For geo-spatial applications, people favor constructed objects, such as roads and buildings, over natural objects, such as vegetation and topography. Natural objects have a clear boundary and are typically of a large extent (e.g., ocean or river). People tend to represent spatial situations in a map-like manner without taking topographic features into account. If the third dimension is considered, then this is limited mostly to individual objects or object groups (e.g., the front view of a house or the silhouette of a town). People keep the shape and structure of their objects simple. Closed boxes and straight lines are the preferred form for representing objects in sketches. Because of their geometric simplicity, sketched objects taken out of context frequently have no meaning of their own. Written annotations are frequently used to describe, augment, or clarify the semantics of sketched objects.

The simplicity of geo-spatial sketches and the fact that certain symbols and sketching pattern are often reused is significant, because it suggests that an automatic analysis and interpretation of geo-spatial sketches is possible. This practicability of sketching, on the other hand, is the basis for using sketches to communicate with computers.

- *Topology matters, while metric and direction refine.*

The survey about the sketching behavior of people confirmed previous observations that people rely primarily on topology to specify spatial object scenes (Egenhofer and Mark 1995). Metric and direction relations between objects are used at a secondary level for refinements. Frequently used topological concepts, besides the predominant *disjoint* relations, are *meet* and *overlap*, while topological relations involving containment are rarely used. Other frequently used spatial concepts among neighboring objects in sketches include parallelity and orthogonality. One third of all objects in our survey had an implicit or explicit direction. The preferred orientation for objects and the sketch is parallel to the vertical or horizontal axis of the sketching surface.

These observations are significant, because they support previous research, indicating that sketches depend primarily on topology. It also shows that people frequently reuse certain spatial or directional concepts. Such knowledge is relevant to interpret sketched scenes automatically.

- *Sketching is an appropriate modality to describe spatial scenes.*

Conventional user interaction methods are frequently not expressive enough for a sophisticated interaction with spatial information. A comparison of various user modalities, considering their suitability for different user tasks, showed that alternative modalities, such as sketching and talking, could positively affect the way people interact with computer systems. Sketching and talking are intuitive, easy to learn, and they have a broad communication bandwidth. These characteristics are important for an interaction with a GIS, because GISs represent complex systems and the composition of users that work with GIS is diverse. While talking is appropriate for a quick, simple, and tool-less user interaction, sketching excels when it comes to describing two-dimensional spatial configurations in an uncomplicated, but expressive way. Therefore, sketching is appropriate to formulate spatial queries that are otherwise difficult to express (e.g., by using a standard spatial query language). The major advantage of a sketched query is that sketching is a direct and visual form of interaction, offering an explicit description of the spatial distribution of objects in a scene. Besides using sketches to query spatial databases, sketching and drawing gestures are also suitable to browse and update spatial databases.

- *Spatial-Query-by-Sketch is a feasible concept of a sketch-based system to query spatial information.*

The prototype has demonstrated that it is possible to automatically analyze and interpret drawn strokes in real-time and to combine them into objects that can be used as building blocks for the generation of the digital sketch. The similarity between different spatial scenes is assessed based on the geometry of objects and spatial characteristics of binary relations between them. This approach and the concept to process a sketch proved to be appropriate as well. The prototype application was successfully employed to rank a set of sketches according to their similarity to a sketched query. These results are relevant, because they show that (1) querying spatial information with sketches is a viable alternative to traditional query methods (Egenhofer 1989), (2) the incorporated spatial formalisms are suitable to compare spatial scenes according to their similarity, and (3) the concept of Spatial-Query-by-Sketch is a practicable approach for a sketch-based system to query spatial information.

8.3 Future Research

The following compilation of research topics has the potential to lead to contributions in the context of systems to query spatial information. Some issues address questions that are complementary to topics discussed in this thesis, while others were raised during the research of this thesis. Most topics include a short description, a set of questions (♦) highlighting interesting aspects, and suggestions (◦) of how individual problems could be approached. The topics are organized into four groups: Conceptual ideas that focus on issues that have not yet been addressed within Spatial-Query-by-Sketch and this thesis (Section 8.3.1), extensions of the prototype implementation (Section 8.3.2), research that the prototype application enables (Section 8.3.3), and a conceptual outline of an integration of sketches with other multimedia data types (Section 8.3.4).

8.3.1 Conceptual Extensions of Spatial-Query-by-Sketch

Much fundamental research has been done concerning the spatial models of Spatial-Query-by-Sketch. However, there is still a lot of room for conceptual extensions. The following sections describe some ideas that have not yet been considered in Spatial-Query-by-Sketch and that aim to extend the capabilities and scope of the application.

8.3.1.1 Incorporation of Multi-modal User Input

The concept of Spatial-Query-by-Sketch is based on sketching to describe spatial situations. However, for the communication of certain types of information it is appropriate to choose different user modalities, such as talking, writing, or typing (Section 3.1.1) (Egenhofer 1996a). Besides issues concerning the *integration* of alternative modalities into a user interface it is important to also consider their *interpretation* and the *synchronization* between individual input channels. An interpretation of alternative modalities is not trivial, since these are often more complex than traditional interaction methods. For instance, a user may choose to switch quickly between different modalities or use multiple modalities (e.g., sketching and talking) at the same time. On the other hand it is also possible that information from different input channels is inconsistent. For instance, if a user is talking about a house while drawing a highway then there is a conflict of semantics that has to be resolved. In this particular case, the user may talk about an object that is already drawn, that will be drawn, or a virtual object that will not be drawn. Possible questions in this context are:

- *How to synchronize different input channels?*
- *How to prioritize different input channels (i.e., what is the importance of individual user modality)?*
- *How to resolve conflicts if input channels contradict each other?*
- *How to effectively interpret, store, and represent alternative user input, such as speech or gestures?*
  - Analyze how people use multiple modalities (Oviatt 1999).
  - In a sketch-and-talk system; find ways to pre-classify objects according to their properties (e.g., geometry). For instance, a box might be a house, a pool, a boundary, but less likely a lake, a forest, or a street.
  - Use ontologies to find the closest match when multiple modalities are being used *(Rodríguez et al. 1999)*. For instance, if a sketched object has been identified as a building (a box was drawn between two streets) and the user specifies the object as the Key Bank, then the system could check for the semantic similarity between the two terms and–in this case–accept the more precise term.

A major goal when examining multi-modal user input is to discover elementary structures or patterns of communication that can be used as elements for the formulation of communication protocols. The motivation is to *interpret*, *synchronize*, *understand*, and *predict* a user's intention.

8.3.1.2 Implicit Objects

Spatial scenes contain explicitly specified objects, but also objects that are derived from drawn objects (Section 4.2.1). Knowledge about implicit objects can play an important role in understanding a sketch. For instance, the intersection of two roads can be the most relevant object in a sketch. However, not every intersection between two objects does necessarily produce a new and relevant object. Research in this context must focus on finding rules that indicate significant implicit objects. People are good in parsing a sketch for important implicit objects. An automated search, on the other hand, is challenging, because knowledge about the geometry *and* the semantics of sketched objects is required.

- *What is the significance of implicit objects in sketches (e.g., their frequency and relevance)?*
- *What are the particular characteristics of a significant implicit object?*
- *What are the relations between implicit objects and sketched objects?*
  - Ask people to mark intersections or other implicit objects in their sketches that they consider important. Compare the interpretation of different human subjects with each other. Is there consent about the set of important implicit objects?
  - *Analyze the parent objects of implicit objects considering such issues as object type or the time when these were drawn.*

8.3.1.3 Network of Semantic Relations

The current implementation of the association graph of the digital sketch considers the spatial neighborhood of sketched objects (*spatial association graph*) (Section 5.3.2). Because sketching is a sequential user modality (Section 3.1.1.3), considering the binary relations between temporal neighbor objects is possible as well (*temporal association graph*) (Section 5.3.1.1). The evaluation between the complete and the reduced association graph has shown that the reduced set of binary relations is sufficient to compare individual sketches according to their similarity (Section 7.3). However, when semantic information of sketched objects, is available then it is possible to consider an additional level of connectivity between objects. This *semantic association graph* may comprise only a small subset of all possible binary object-object relations; however, this graph is likely to contain valuable information, because it connects only those objects that have a semantically meaningful relation. The graph can also be seen as a form of *knowledge representation* of a sketch (Shapiro and Rapaport 1992). An example of a semantic association graph is the connectivity chart of an airline, which interconnects all destinations.

- *How to construct a semantic association graph and what are the criteria to connect object pairs?*
- *Is there a need to interconnect multiple objects (grouping)?*
- *Is it possible or necessary to differentiate multiple semantic association graphs?*
- *What is the relation between the spatial, temporal, and semantic association graphs?*
  - Ask human subjects to connect those objects of a mockup scene that they consider standing in an important interrelation. Is there any correlation between individual interpretations of a spatial scene? Try to come up with a rationale why people connect certain objects. Consider object properties, such as object type, purpose, or location and object-object relations for this purpose.

8.3.1.4 Time and Change

The digital sketch considers the static representation of a spatial scene. However, for certain applications, such as querying a dynamic spatial scene, it is appropriate to extend this notion so that objects are allowed to change their state and location over time. A query such as “Show me all police cars that have followed this particular route last Saturday, in this direction” in conjunction with a sketch is an example for such a query. Besides an object’s location other object properties can change over time as well. An object can, for instance, split, unite with another object, or cease to exist (Hornsby 1999). Questions in this context include:

- *How to integrate temporal change into the model of the digital sketch?*
- *How to capture, store, and represent properties of sketched objects and spatial relations that change over time?*

- *How to intuitively describe a temporal event in a sketch (e.g., by moving a sketched object along a route)?*
  - Ask people to describe dynamic events graphically with sketches.
  - Alternatively, allow people to express themselves verbally while sketching.
  - Analyze how people describe changes and what modalities they prefer.
  - Think of tools that support users in describing temporal changes. For instance, use a slider bar to animate a sketched scene or use layers or symbols to represent different states.

8.3.1.5 Support for Database Systems

Querying a spatial scene with the current implementation involves a linear search that compares a digital sketch (sketched query) with all digital sketches in a data repository (individual spatial scenes stored as ASCII files in a directory). This approach works fine for a prototype implementation, but it does not scale up for a real world application. There are two possible scenarios for a real world database: (1) the database consists of a large number of individual data records, which this is similar to the current implementation, (2) the database consists of a large continuous dataset containing spatial information. Both settings require different approaches for an efficient processing.

A key problem, in this context, is the association of corresponding objects in the sketched query and the database, which is the base for a similarity assessment between two spatial configurations. Because there are more potential matches for a queried object in a continuous database than in an individual data record (assuming an equal distribution of objects), this issue is more critical for large continuous databases. To avoid linear or brute-force searches that test all possible combinations, methods have to be found that allow *smart queries*. These queries rely on metadata (derived from the data in the database) and on sophisticated indexes that allow an application to focus on a small subset of all possible matches. In this context we argue that it is advantageous to store information about the neighborhood of objects and their immediate relations with other objects besides pure object-focused information. A consideration of spatial, temporal, or contextual neighborhoods applied to an entire database appears to be a possible approach.

- *How to efficiently index spatial objects and relations?*
- *How to intelligently associate a set of objects in a query scene with a set of objects in a large continuous database?*
- *How to efficiently query a set of individual data records?*
- *How to efficiently query a continuous database?*

8.3.2 Software Extensions of the Prototype

Another set of future research questions deals with implementation-specific extensions of the software protoype. These extensions attempt to enhance the functionability of the user interface, increase the expressiveness of the digital sketch, and aim to improve the overall capabilities of the system. Most of the presented extensions are based on existing research or research in progress. Some extensions were introduced in the previous section, while others involve an additional research effort, however in a smaller scope.

8.3.2.1 Verbal Annotations

A multi-modal user interface should allow users to interact with a system through multiple input channels, including voice, where this is appropriate. Adding this additional channel poses some great challenges, such as how to synchronize verbal input with other input sources or how to react when inconsistencies arise (Section 8.3.1.1). If more than simple words are involved, then these challenges are more complex, because verbal statements have to be parsed and interpreted (Scha 1988). Since language can contain a lot of complementary information, this effort seems to be well justified.

8.3.2.2 Line and Point Objects

The current implementation of the digital sketch approximates lines as thin regions, while objects of type point are not yet considered. An integration of additional object classes requires an adaptation and extension of the formalisms describing the spatial relations between objects (Shariff 1996). Adding points and lines to the digital sketch introduces multiple representations of sketched objects (Bertolotto *et al.* 1995). For instance, a virtual object, such as a town specified only by a name, can either be a point or region object. This changes the set of possible topological relations with neighboring objects.

- ♦ *How to cope with different object representations?*
- ♦ *In the case of multiple representations, how to translate relations of one type into another (e.g., region-region relation into a line-region relation)?*
- ◦ Extend the digital sketch for points and lines.
- ◦ Implement formalisms for all resulting relation combinations.
- ◦ Investigate how to treat ambiguities, that is, how to handle objects that can be stored as different types.

8.3.2.3 Geometric Object Descriptors

The digital sketch characterizes sketched objects using a small set of geometric attributes and simplifies them at different levels (e.g., outline or tilted MBR); however, additional geometric descriptors are necessary to better classify sketched objects (e.g., for an efficient indexing; Section 8.3.1.5). Additional geometric descriptors could include shape or other object properties.

- ◦ Ask human subjects to rank a set of similar objects (with different visual properties) according to their similarity to a query object.
- ♦ *What object properties are most relevant for this classification? Such an analysis may lead to a classification of object attributes (e.g., the shape of an object is more important than its size).*
- ◦ Find a set of object characteristics that is suitable to describe sketched objects.

- ◦ Investigate how relevant individual object properties are.
  - *Is this classification dependent on the context?*
  - *What is the influence on this assessment when people have to rank groups of objects (e.g., if objects are in a specific spatial relation to each other)?*
  - *What is more important, object properties or the spatial configuration between these objects?*

8.3.2.4 Alternative Formalism for Direction Relations

The currently used method to capture the direction relation between two objects is sensitive to rotation, that is, the method works only for scenes that are rotated in increments of 90° and the directional component is only cognitively feasible for objects that are close together. Current research efforts are addressing these issues (Goyal and Egenhofer in press).

- ◦ Implement the new direction assessment method as a configurable option.
- ◦ Compare the approach with the current direction assessment method.

8.3.2.5 Semantics of Objects

The geometry of objects and knowledge about their spatial configuration are appropriate to describe a scene at an abstract level. In order to understand the meaning of a spatial scene, however, it is necessary to be aware of the semantics of involved objects. Such knowledge is also advantageous to minimize the number of potential candidates (objects) in a database that have to be compared with objects in a sketched query (Section 8.3.1.5). To attach semantic information to sketched objects there are basically two ways: (1) the system can automatically derive the semantics from object attributes, such as an object's shape or its symbolic representation, or (2) the semantics have to be specified explicitly (e.g., by typing, handwriting writing, or talking). How to capture the semantics of objects and how to assess the similarity between different semantic expressions are currently investigated research questions (Rodríguez *et al.* 1999).

- Extend the digital sketch so that semantic object information can be incorporated.
- Implement a mechanism that allows a similarity assessment between different semantic expressions.

8.3.2.6 Interactive Result Browsing

An interaction on top of the results of a spatial query is currently limited to changing the weights of the components of the sketched query (objects and spatial relations) and re-assigning objects that were incorrectly associated. However, the success of the World-Wide-Web indicates that an *enhanced interaction on top of retrieved results* is a popular and promising concept. An efficient method to present initially retrieved results is to view the results within a suitable context. Appropriate forms of representation for this purpose are, for instance, maps, orthophotos, or images. Suitable method for browsing results can involve Pad++-like operations (Benderson *et al.* 1996) or hyperlinks that zoom into other levels of detail or that open related documents (graphics or text) (Stone *et al.* 1994). A scenario like this, which allows a user to browse for additional information, requires an elaborate linking mechanism that connects database objects at different levels with each other (Egenhofer 1997b). In this context it has to be evaluated in how far today's multimedia technologies are suitable for browsing a spatial database.

- Investigate the requirements for browsing a spatial database.
- Explore the capabilities of today's multimedia technologies.
- Combine the two worlds and extend the concept where necessary.

8.3.3 Using the Prototype as a Platform for Future Research

The main purpose of the prototype implementation was to create a solid test bed for the evaluation of the hypothesis of this thesis. However, due to its comprehensive scope, the prototype can serve as a platform for further research as well. Such applications include using the prototype as a base to implement and test additional spatial formalisms, as well as using the implementation in its present configuration to conduct experimental research.

8.3.3.1 Automated Human Subject Testing

In Chapter 4 we investigated the sketching behavior of people. Since manual methods were used to analyze individual sketches, it is desirable to verify the results by an automatic analysis. Besides conducting an automated survey with a similar setup, other relevant parameters can be assessed as well. Recording the surveyed human subjects with video cameras (Hewett 1997) provides additional insights about an interaction between user and computer, because such recordings capture supplementary information, such as gestures or eventual verbal interaction with the system.

- *Is there a difference between the results of the automated and the manual survey?*
- *How frequently do people use verbal expressions during the sketching process?*
- *Do people sketch and talk simultaneously?*
- *Are sketched and verbal input synchronized?*
- *How frequently are drawn objects modified once they have been drawn?*
- *How do people cope with the mechanism that automatically aggregate strokes to objects?*
- *How successful is the automatic object association, and how precise is the abstraction mechanism for objects?*

8.3.3.2 Automated vs. Manual Similarity Assessment of Spatial Scenes

By using the implemented similarity assessment methods, the prototype can rank a set of sketches according to their scene similarity with a sketched query. Because the methods used for this assessment are based on several formalisms that were developed individually, a calibration of the formalisms' weights is crucial. For this purpose the automated ranking of a set of sketches (by the prototype) can be compared with the manual ranking of the same set of sketches (by human subjects).

- *How do people weight individual components of the digital sketch (e.g., topology vs. metric vs. direction vs. completeness)?*

- *Is the assessment of the relevance of individual components consistent for different users and for different tasks?*

8.3.3.3 Traditional Pens vs. Electronic Pens

Sketching on paper with a pen and sketching with an electronic pen on a computer are conceptually similar, but not identical forms of interaction. It is, therefore, of interest to analyze how making an electronic sketch compares to making a conventional sketch. Such an investigation could involve also a comparison of different sizes of the sketching surface (e.g., comparing a PalmPilot-sized device with a standard letter-sized device).

- *Are electronic sketches different from analogue sketches (e.g., considering number of strokes per object, number of objects per sketch, orientation of objects, or use of space)?*
  - Ask human subjects to draw a conventional version (on plain paper) and an electronic version (using the prototype) of the same sketch. To obtain classifiable results, the two sketches must not be drawn immediately one after each other. The goal is to define the break such that the subjects remember the concept of the sketch, but not the details of its creation.

8.3.3.4 Testing new Spatial Formalisms

Besides using the prototype as a test bed for already implemented theories, it can also be used to test newly developed spatial formalisms. In this context it is feasible to compare new with already implemented formalisms (e.g., if they describe the same spatial characteristics) or to analyze the implications of new formalisms on the results of a sketched query.

8.3.4 Integration of Sketches and other Multimedia Data Types

Sketches and other descriptive representations of reality, such as images or paper maps can be viewed as analog models of instances in reality. Multimedia data types include also digital models of reality (e.g., a digital image). An application that translates one form of representation into another is as a converter. The prototype implementation that was developed in the scope of this thesis is, therefore, essentially an *analog-to-digital sketch converter*. There exist different methods to convert analog models into corresponding digital models so that they can be processed with a computing system. Some of these approaches are limited to mapping analog information at a certain resolution (e.g., scanning a photograph), while other approaches put an additional interpretation effort into the analysis of analog information (e.g., Spatial-Query-by-Sketch or SNEPS (Srihari and Rapaport 1989; Shapiro and Rapaport 1992)). Figure 8.1 shows some analog and digital models that describe reality, and possible conversion methods.

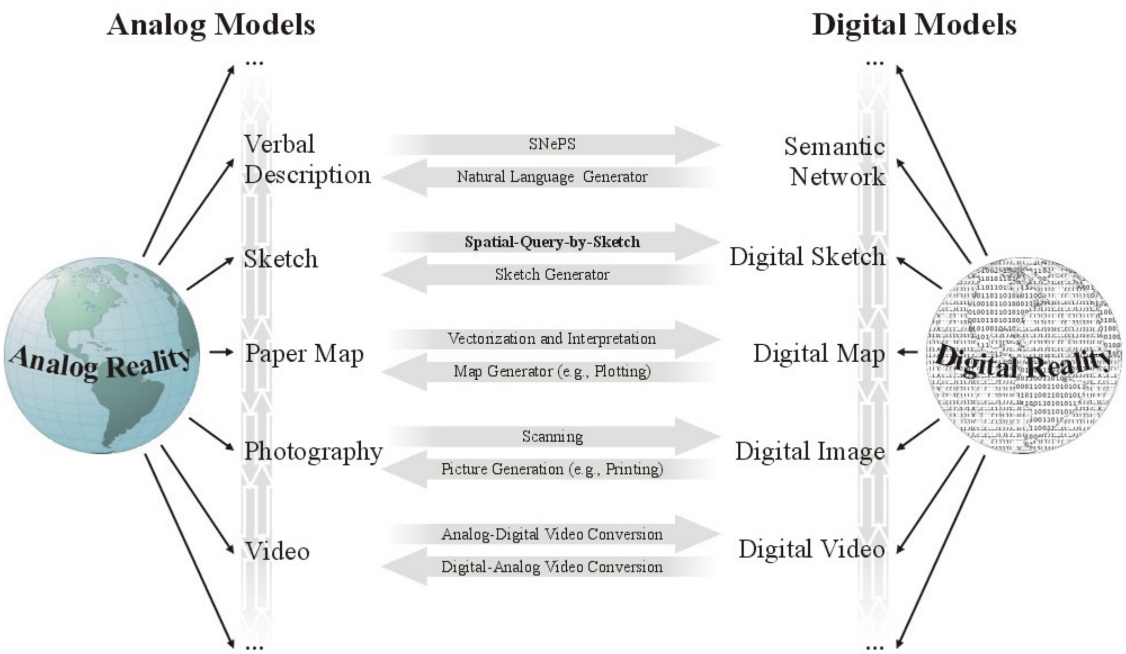


Figure 8.1 Interfaces between different multimedia data types.

Besides conversion mechanisms that translate corresponding models into each other (i.e., horizontal conversion), there are those converters that translate between different digital models (i.e., vertical conversion). An example of such a vertical converter is a mechanism that translates a digital image into a digital map. Depending on the complexity of individual models and on the type of information they represent (e.g., pixels in a digital image have no knowledge about their neighbor pixels, while entities in a object-oriented models, such as a digital sketch, pertain higher level information) such a conversion can be more or less complex. The rationale behind Figure 8.1 is that in order to portray the real world (analog reality) in a form that is computationally accessible it is essential to provide a *set of horizontal and vertical conversion mechanisms*. The goal of such a system is to create a connected network with individual models as nodes and conversion mechanisms as edges, so that every model can be translated into any other model. However, this approach does not necessarily require all models to be directly interconnected, because it is feasible to derive individual models indirectly from each other.

Research in the context of such an *integrated multimedia translation system* can focus on individual conversion mechanisms that serve as a base for more complex conversions. Possible applications are, for instance, the generation of a paper map based on a photograph or the translation of a sketch into a natural language statement. However, also futuristic scenarios involving time, space, and context, such as the translation of a book in to a movie, are imaginable.

References

P. Agouris, J. Carswell, and A. Stefanidis (1999) An Environment for Content-Based Image Retrieval from Large Spatial Databases. Photogrammetry and Remote Sensing *54: 263-272.*

P. Agouris and T. Schenk (1996) Automated Aerotriangulation Using Multiple Image Multipoint Matching. Photogrammetric Engineering and Remote Sensing *62(6): 703-710.*

J. Allen (1983) Maintaining Knowledge about Temporal Intervals. Communications of the ACM *26(11): 832-843.*

J. Allen (1995) Natural Language Understanding. *The Benjamin/Cummings Publishing Company, Inc., Redwood City, CA.*

F. Aurenhammer (1991) Voronoi Diagrams-A Survey of a Fundamental Geometric Data Structure. ACM Computing Surveys *23(3): 345-406.*

R. Barber, M. Flickner, J. Hafner, D. Lee, W. Niblack, D. Petkovic, J. Ashley, T. McConnell, J. Ho, J. Jang, D. Berkowitz, P. Yanke, M. Vo, D. Haas, D. Lassing, S. Tate, A. Chang, P. van Houten, J. Chang, T. Peterson, D. Lautrell, M. Snedden, P. Faust, C. Matteucci, M. Raynor, R. Peters, W. Beck, and J. Witsett (1994) Ultimedia Manager: Query by Image Content and its Applications. IEEE Computer Society*: 424-429.*

B. Benderson and J. Hollan (1994) Pad++: A Zooming Graphical Interface for Exploring Alternative Interface Physics. in: Proc. 7th Annual Symposium on User Interface Software and Technology (UIST'94*. pp. 39-48, ACM Press, New York.*

B. Benderson, J. Jollan, K. Perlin, J. Meyer, D. Bacon, and G. Furnas (1996) Pad++: A Zoomable Graphical Sketchpad for Exploring Alternate Interface Physics. Journal of Visual Languages and Computing *7(1): 3-31.*

M. Bertolotto, L. De Floriani, and P. Marzano (1995) A Unifying Framework for Multilevel Description of Spatial Data. in: A. Frank and W. Kuhn (Eds.), Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '95, Semmering, Austria. Lecture Notes in Computer Science *988, pp. 259-278, Springer-Verlag, Berlin.*

M. Blades (1990) The Reliability of Data Collected from Sketch Maps. Journal of Environmental Psychology *10(4): 327-339.*

A. Blaser (1997) User Interaction in a Sketch-Based GIS User Interface. in: A. U. F. Stephen C. Hirtle (Ed.) International Conference COSIT '97 (Poster Session), *Laurel Highlands, PA, pp. 505.*

A. Blaser (1998) Geo-Spatial Sketches*. National Center of Geographic Information and Analysis (NCGIA), University of Maine, Orono, Technical Report TR98-1.*

A. Blaser (1999a) Prototype Application Sketcho. *National Center of Geographic Information and Analysis (NCGIA), University of Maine, Orono, Technical Report TR99-1.*

A. Blaser (1999b) Users Guide for Sketcho. *National Center of Geographic Information and Analysis (NCGIA), University of Maine, Orono, Technical Report TR99-2b / 2nd Rev.*

A. Blaser, M. Sester, and M. Egenhofer (2000) Visualization in an Early Stage of the Problem Solving Process in GIS. Computers & Geosciences *26(1).*

T. Bruns and M. Egenhofer (1996) Similarity of Spatial Scenes. in: M.-J. Kraak and M. Molenaar (Eds.), Seventh International Symposium on Spatial Data Handling, *Delft, The Netherlands, pp. 31-42.*

W. Buxton (1986) There's More to Interaction Than Meets the Eye: Some Issues in Manual Input. in: D. Norman and S. Draper (Eds.), User Centered Systems Design*. pp. 319-338, Lawrence Erlbaum Associates, Hillsdale, NJ.*

D. Calcinelli and M. Mainguenaud (1994) Cigales, a Visual Query Language for a Geographical Information System: the User Interface. Journal of Visual Languages and Computing *5(2): 113-132.*

S. Card, T. Moran, and A. Newell (1980) The Keystroke-Level Model for User Performance Time with Interactive Systems. Communications of the ACM *23(7): 396-410.*

D. Chamberlin, M. Astrahan, K. Eswaran, P. Griffiths, R. Lorie, J. Mehl, P. Reisner, and B. Wade (1976) SEQUEL 2: A Unified Approach to Data Definition, Manipulation, and Control. IBM Journal of Research and Development *20(6): 560-575.*

S. Chang, Q. Shi, and C. Yan (1987) Iconic Indexing by 2-D Strings. IEEE Transaction Pattern Analysys Machine Intelligence *9(3): 413-428.*

M. Chignell and P. Hancock (1988) Intelligent Interface Design. in: M. Helander (Ed.) Handbook of Human-Computer Interaction*. pp. 969-991, Elsevier Science Publishers B.V. (North-Holland), Amsterdam.*

C. Chui (1992) An Introduction to Wavelets*. Academic Press Inc., Boston, MA.*

W. Citrin and M. Gross (1996) Distributed Architectures for Pen-Based Input and Diagram Recognition. in: Proceedings of the ACM, AVI '96*, Gubbio, Italy, pp. 132-140.*

A. Collins (1979) Children's Language and Communication. in: A. Collins (Ed.) The Minnesota Symposia on Child Psychology*, Minnesota.*

J. Coppock and D. Rhind (1991) The History of GIS. in: D. Maguire and M. Goodchild (Eds.), Geographical Information Systems*. pp. 21-43, Longman, London.*

J. Corridoni, A. Del Bimbo, and S. De Magistris (1996a) Image Query by Semantical Colour Content. in: AVI '96*, Gubbio Italy, pp. 213-222.*

J. Corridoni, A. Del Bimbo, S. De Magistris, and E. Vicario (1996b) A Visual Language for Color-Based Painting Retrieval. in: Visual Languages 96*, pp. 68-75.*

A. Cypher (1986) The Structure of Users' Activities. in: D. Norman and S. Draper (Eds.), User Centered Systems Design*. pp. 243-264, Lawrence Erlbaum Associates, Hillsdale, NJ.*

W. Daniel (1978) Applied Nonparametric Statistics*. Houghton Mifflin Company, Boston, MA.*

M. de Berg, M. van Kreveld, M. Overmars, and O. Schwarzkopf (1997) Computational Geometry*. Springer-Verlag, Berlin.*

A. Del Bimbo, M. Campanai, and P. Nesi (1992) 3-D Visual Query Language for Image Databases. Journal of Visual Languages and Computing *3(3): 257-272.*

A. Del Bimbo, M. Campanai, and P. Nesi (1993) A Three-Dimensional Iconic Environment for Image Database Querying. IEEE Transaction on Software Engineering *19(10): 997-1011.*

A. Del Bimbo and P. Pala (1997) Visual Image Retrieval by Elastic Matching of User Sketches. IEEE Transactions on Pattern Analysis and Machine Intelligence *19(2): 121-132.*

A. Del Bimbo, P. Pala, and S. Santini (1994) Visual Image Retrieval by Elastic Deformation of Object Sketches. in: IEEE Symposium on Visual Languages, *St. Louis, MO, pp. 216-223.*

R. DeVore, B. Jawerth, and B. Lucier (1992) Image Compression Through Wavelet Transform Coding. IEEE Transactions on Information Theory *38(2): 719-746.*

F. Di Loreto, F. Ferri, F. Massari, and M. Rafanelli (1995) A Visual Object-Oriented Query Language for Geographic Information Systems. in: N. Revell and A. Tjoa (Eds.), Database and Expert Systems Applications: 6th International Conference, DEXA'95, London, United Kingdom*.* Lecture Notes in Computer Science *978, pp. 103-113, Springer-Verlag, New York.*

F. Di Loreto, F. Ferri, F. Massari, and M. Rafanelli (1996) A Pictorial Query Language for Geographic Databases (PQL). in: AVI '96*, Gubbio, Italy, pp. 233-244.*

S. Draper (1996) Course Notes on User Interaction Styles. *University of Glasgow, Glasgow, Technical Report.*

M. Egenhofer (1989) Spatial Query Languages*. Ph.D, University of Maine, Orono, ME.*

M. Egenhofer (1990) Interaction with Geographic Information Systems via Spatial Queries. Journal of Visual Languages and Computing *1(4): 389-413.*

M. Egenhofer (1992) Why not SQL! International Journal of Geographical Information Systems *6(2): 71-85.*

M. Egenhofer (1996a) Multi-Modal Spatial Querying. in: M.-J. Kraak and M. Molenaar (Eds.), Seventh International Symposium on Spatial Data Handling, *Delft, The Netherlands, pp. 12B.1-15.*

M. Egenhofer (1996b) Spatial-Query-by-Sketch. in: M. Burnett and W. Citrin (Eds.), VL '96: IEEE Symposium on Visual Languages*. pp. 60-67, IEEE Computer Society, Boulder, CO.*

M. Egenhofer (1997a) Query Processing in Spatial-Query-by-Sketch. Journal of Visual Languages and Computing *8(4): 403-424.*

M. Egenhofer (1997b) Smart Maps (Keynote Address). in: 5th ACM Workshop on Geographic Information System*, Las Vegas, NV.*

M. Egenhofer and K. Al-Taha (1992) Reasoning About Gradual Changes of Topological Relationships. in: A. Frank, I. Campari, and U. Formentini (Eds.), Theories and Methods of Spatio-Temporal Reasoning in Geographic Space, Pisa, Italy. Lecture Notes in Computer Science *639, pp. 196-219, Springer-Verlag, Berlin.*

M. Egenhofer and R. Franzosa (1991) Point-Set Topological Spatial Relations. International Journal of Geographical Information Systems *5(2): 161-174.*

M. Egenhofer and J. Herring (1990) A Mathematical Framework for the Definition of Topological Relationships. in: K. Brassel and H. Kishimoto (Eds.), Fourth International Symposium on Spatial Data Handling*, Zurich, Switzerland, pp. 803-813.*

M. Egenhofer and J. Herring (1991) Categorizing Binary Topological Relationships Between Regions, Lines, and Points in Geographic Databases. *Department of Surveying Engineering, University of Maine, Orono, ME.*

M. Egenhofer and W. Kuhn (1998) Beyond Desktop GIS. in: GIS PlaNET, *Lisbon.*

M. Egenhofer and W. Kuhn (1999) Interacting with GIS. in: P. Longley, M. Goodchild, M. Maguire, and D. Rhind (Eds.), Geographical Information Systems: Principles, Techniques, Management, and Applications. *pp. 401-412, Geoinformation International, London.*

M. Egenhofer and D. Mark (1995) Naive Geography. in: A. Frank and W. Kuhn (Eds.), Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '95, Semmering, Austria. Lecture Notes in Computer Science *988, pp. 1-15, Springer-Verlag, Berlin.*

R. Elmasri and S. Navathe (2000) Fundamentals of Database Systems. *Addison-Wesley, Reading, MA.*

I. Encyclopedia Britannica (1996) Encyclopedia Britannica. *1994-1997 Encyclopedia Britannica, Inc, Chicago.*

S. Feiner (1992) Virtual Worlds for Visualizing Information. in: T. Catarci, M. Costabile, and S. Levialdi (Eds.), Advanced Visual Interfaces, Proceedings of the International Workshop AVI '92. *36, pp. 3-11, World Scientific, Singapore.*

M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker (1995) Query by Image and Video Content: The QBIC System. IEEE Computer *28(9): 23-32.*

J. Florence (1997) Predicting Frequencies of Topological Relations in Geographic Datasets. *M.S. Thesis, University of Maine, Orono.*

S. Fortune (1985) Sweepline Algorithm for Voronoi Diagrams. Algorithmica *2: 153-174.*

S. Fortune (1992) Voronoi Diagrams and Delaunay Triangulations. in: D. Du and F. Hwang (Eds.), Euclidean Geometry and Computers*. pp. 193-233, World Scientific Publishing Co.*

A. Frank and S. Timpf (1994) Intelligent Zooming. Computers and Graphics *18(6): 823-830.*

K. Gabriel and R. Sokal (1969) A New Statistical Approach to Geographic Variation Analysis. Systematic Zoology *18: 259-278.*

T. Geisel and A. Geisel (1960) Green Eggs and Ham*. Beginner Bokks Inc., New York.*

J. Gibbons (1985) Nonparametric Methods for Quantitative Analysis (2nd Edition). *American Science Press, Inc, Columbus, OH.*

A. Goldberg (1988) A History of Personal Workstations. *Addison-Wesley Publishing Company, New York.*

A. Goldberg and A. Kay (1976) Smalltalk-72 Instructional Manual*. Xerox PARC, Palo Alto, California, Technical Report Technical Report SSL-76-6.*

A. Goldberg and D. Robson (1983) Smalltalk-80*. Addison-Wesley Publishing Company, Reading, MA.*

G. Goldschmidt (1991) The Dialectics of Sketching. Creativity Research Journal *4(2): 123-143.*

M. Goodchild (1987) A Spatial Analytical Perspective on Geographical Information Systems. International Journal of Geographical Information Systems *1(4): 327-334.*

J. Gould and C. Lewid (1985) Design for Usability: Key Principles and What Designers Think. in: B. Hackle (Ed.) INTERACT '84: First Conference on Human-Computer Interaction*, Amsterdam, North Holland.*

R. Goyal and M. Egenhofer (in press) Cardinal Directions between Extended Spatial Objects. IEEE Transactions on Knowledge and Data Engineering.

J. Greenstein and L. Arnaut (1998) Input Devices. in: M. Helander (Ed.) Handbook of Human-Computer Interaction*. pp. 495-516, Elseviere Science Publishers B. V. (North-Holland), Amsterdam.*

M. Gross (1994a) Recognizing and Interpreting Diagrams in Design. in: Advanced Visual Interfaces '94*, pp. 89-94.*

M. Gross (1994b) Stretch-A-Sketch: A Dynamic Diagrammer. in: A. Ambler and T. Kimura (Eds.), IEEE Symposium on Visual Languages*, St. Louis, MO, pp. 232-238.*

M. Gross (1996) The Electronic Cocktail Napkin-Computer Support for Working with Diagrams. Design Studies *17(1): 53-69.*

V. Haarslev (1996) Using Description Logic for Reasoning about Diagrammatical Notations. in: L. Padgham (Ed.) Proceedings of the International Workshop on Description Logics*, Cambridge, MA, pp. 124-128.*

V. Haarslev and R. Möller (1997) SBox: A Qualitative Spatial Reasoner. in: L. Ironi (Ed.) Proceedings, 11th IEEE Symposium on Qualitative Reasoning*, Cortona, Italy, pp. 105-113.*

V. Haarslev and M. Wessel (1997) Querying GIS with Animated Spatial Sketches. in: Proceedings of the 13th IEEE Symposium on Visual Languages 1997*, Capri, Italy, pp. 197-204.*

M. Helander, Ed. (1988) Handbook of Human-Computer Interaction*. Elsevier Science Publishers (North Holland), Amsterdam.*

T. Hewett (1997) International Conference COSIT 97 (Doctorial Consortium)*, Laurel Highlands, PA.*

T. Hewett, R. Baecker, S. Card, T. Carey, J. Gasen, M. Mantei, G. Perlman, G. Strong, and W. Verplank (1992) Report of the ACM Special Interest Group on Computer-Human Interaction (SIGCHI)*. ACM / SIGCHI, Salem, MA, Technical Report available from http://www.acm.org/sigchi/cdg/cdg2.html.*

K. Hirata and T. Kato (1992) Query by Visual Example—Content-Based Image Retrieval. in: A. Pirotte, C. Delobel, and G. Gottlob (Eds.), Advances in Database Technology—EDBT '92, 3rd International Conference on Extending Database Technology, Vienna, Austria. Lecture Notes in Computer Science *580, pp. 56-71, Springer-Verlag, Berlin.*

K. Hirata and T. Kato (1993) Rough Sketch-Based Image Information Retrieval. NEC Research and Development *34(2): 263-273.*

K. Hornsby (1999) Identity-Based Reasoning About Spatio-Temporal Change. *Ph.D. Thesis, University of Maine, Orono.*

E. Hutchins, J. Hollan, and D. Norman (1986) Direct Manipulation Interfaces. in: D. Norman and S. Draper (Eds.), User Centered Systems Design*. Lawrence Erlbaum Associates, Hillsdale, NJ.*

J. Jackson (1990) Visualization of Metaphors for Interaction with Geographic Information Systems. *Master's thesis, University of Maine.*

C. Jacobs, A. Finkelstein, and D. Salesin (1995) Fast Multiresolution Image Querying. in: ACM SIGGRAPH, *Los Angeles, CA, pp. 277-286.*

J. Jaromczyk and G. Toussaint (1992) Relative Neighborhood Graphs and their Relatives. Proceedings of the IEEE *80(9): 1502-1517.*

J. Johnson (1995) A Comparison of User Interfaces for Panning on a Touch-Controlled Display. in: CHI '95, *Denver, CO, pp. 218-225.*

T. Kato (1996) Query by Visual Example. Accessed*: 11/10/1999, http://www.etl.go.jp:8080/etl/taiwa/HumanMedia/HM-ETL/welcome.html.*

A. Kay and A. Goldberg (1977) Personal Dynamic Media. IEEE Computer *10(3): 31-42.*

P. Kelly, T. Cannon, and D. Hush (1995) Query by Image Example: The CANDID Approach. SPIE Storage and Retrieval for Image and Video Databases III *2420: 238-248.*

D. Kornbrot (1990) The rank difference test: A new and meaningful alternative to the Wilcoxon signed ranks test for ordinal data. Journal of Mathematical and Statistical Psychology *43: 241-264.*

W. Kuhn (1992) Paradigms of GIS Use. in: D. Cowen (Ed.) Fifth International Symposium on Spatial Data Handling. *pp. 91-103, Charleston, SC.*

W. Kuhn (1993) Metaphors Create Theories for Users. in: A. Frank and I. Campari (Eds.), Spatial Information Theory, European Conference COSIT '93, Marciana Marina, Elba Island, Italy. Lecture Notes in Computer Science *716, pp. 366-376, Springer-Verlag, New York.*

W. Kuhn and A. Frank (1991) A Formalization of Metaphors and Image-Schemas in User Interfaces. in: D. Mark and A. Frank (Eds.), Cognitive and Linguistic Aspects of Geographic Space. *pp. 419-434, Kluwer Academic Publishers, Dordrecht.*

G. Lakoff and M. Johnson (1980) Metaphors We Live By. *University of Chicago Press, Chicago, IL.*

P. Lankford (1969) Regionalization: Theory and Alternative Algorithms. Geographical Analysis *1: 196-212.*

J. Larkin and H. Simon (1987) Why a Diagram is (Sometimes) Worth Ten Thousand Words. Cognitive Science *11: 65-99.*

R. Laurini and D. Thompson (1992) Fundamentals of Spatial Information Systems. *Academic Press, London.*

G. Leach, G. Al-Qaimari, M. Grieve, J. N., and C. McKay (1997) Elements of a Three-Dimensional Graphical User Interface. in: INTERACT'97*, Sydney, Australia.*

Y. Lee and F. Chin (1995) An Iconic Query Language for Topological Relationships in GIS. International Journal of Geographical Information Systems *9(1): 25-46.*

M. Levick (1986) Mommy, Daddy, Look what I'm Saying: What Children are Telling through their Drawings. *M. Evans, NT.*

M. Levick (1998) See What I'm Saying. *Islewest Publishing, Dubuque, IA.*

T. Lewis (1995) Living in Real Time, Side A. IEEE Computer *28(9): 8-10.*

G. Liotta (1996) Low Degree Algorithms for Computing and Checking Gabriel Graphs. *Brown University, CS Dept., Providence, RI, Technical Report CS-96-28.*

J. Lipscomb (1991) A Trainable Gesture Recognizer. Pattern Recognition *24(9): 895-907.*

P. Longley, M. Goodchild, and D. Maguire (1999) Geographical Information Systems: Principles, Techniques, Applications and Management.

K. Lynch (1960) The Image of a City. *MIT Press, Cambridge, MA.*

W. Maaß (1995) How Spatial Information Connects Visual Information and Natural Language Generation in Dynamic Environments: Towards a Computational Model. in: A. Frank and W. Kuhn (Eds.), Spatial Information Theory—A Theoretical Basis for GIS, International Conference COSIT '95, Semmering, Austria. Lecture Notes in Computer Science *988, pp. 223-240, Springer-Verlag, Berlin.*

D. Maguire, M. Goodchild, and D. Rhind, Eds. (1991) Geographical Information Systems. *Longman, London.*

M. Mainguenaud and M.-A. Portier (1990) Cigales: A Graphical Query Language for Geographical Information Systems. in: K. Brassel and H. Kishimoto (Eds.), Fourth International Symposium on Spatial Data Handling, *Zurich, Switzerland, pp. 393-404.*

D. Mark and M. Egenhofer (1994) Modeling Spatial Relations Between Lines and Regions: Combining Formal Mathematical Models and Human Subjects Testing. Cartography and Geographic Information Systems *21(3): 195-212.*

R. Mehrotra and J. Gary (1993) Feature-Based Retrieval of Similar Shapes. in: Ninth International Conference on Data Engineering*, Vienna, Austria, pp. 108-115.*

R. Mehrotra and J. Gray (1995) Similar-Shape Retrieval in Shape Data Management. IEEE Computer *28(9): 57-62.*

B. Mel, S. Omohundro, A. Robinson, S. Skiena, K. Thearling, S. Wolfram, and L. Young (1988) Tablet: The Personal Computer in the Year 2000. Communications of the ACM*: 638-646.*

B. Meyer (1993) Beyond Icons—Towards New Metaphors for Visual Query Languages for Spatial Information Systems. in: International Workshop on Interfaces to Database Systems*, Glasgow, Scotland, pp. 113-135.*

B. Meyer (1994) Pictorial Deduction in Spatial Information Systems. in: A. Ambler and T. Kimura (Eds.), IEEE Symposium on Visual Languages*, St. Louis, MO, pp. 23-39.*

Microsoft (1999) Encarta Encyclopedia*. Microsoft Corporation, Redmond, WA.*

U. Montanari (1968) A method for obtaining skeletons using a quasi-Euclidean distance. Journal of the ACM *15(4): 600-624.*

F. Mosteller and R. Rourke (1973) Sturdy Statistics*. Addison-Wesley Publishing Company, Menlo Park, CA.*

S. Mountford (1992) What can Users Tell us about User Interface? in: T. Catarci, M. Costabile, and S. Levialdi (Eds.), Advanced Visual Interfaces, Proceedings of the International Workshop AVI '92*. 36, pp. 103-107, World Scientific, Singpore.*

B. Myers (1996) A Brief History of Human Computer Interaction Technology*. Carnegie Mellon University School of Computer Science, Pittsburgh, Technical Report CMU-CS-96-163.*

J. Neal, K. Bettinger, J. Byoun, Z. Dobes, and C. Thielman (1988) An Intelligent Multi-Media Human-Computer Dialogue System. in: Workshop on Space Operations Automation and Robotics (SOAR 88)*, Dayton, OH, pp. 69-80.*

J. Neal and S. Shapiro (1988) Intelligent Multi-Media Interface Technology. in: Architectures for Intelligent Interfaces: Elements and Prototypes*, Monterey, CA, pp. 69-80.*

J. O'Rourke (1993) Computational Geometry in C. *Cambridge University Press, Cambridge.*

S. Oviatt (1997) Multimodal interactive maps: Designing for human performance,. Human-Computer Interaction*: 93-129.*

S. Oviatt (1999) Ten Myth of Multimodal Interaction. Communication of the ACM *42(11): 74-81.*

R. Owens (1996) Language Development: An Introduction*. Allyn and Bacon, New York.*

D. Papadias, T. Sellis, Y. Theodoridis, and M. Egenhofer (1995) Topological Relations in the World of Minimum Bounding Rectangles: A Study with R-Trees. Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*: 92-103.*

F. Preparata and M. Shamos (1985) Computational Geometry. *Springer-Verlag, New York.*

A. Rodríguez, M. Egenhofer, and R. Rugg (1999) Assessing Semantic Similarities Among Geospatial Feature Class Definitions. in: A. Vckovski, K. Brassel, and H.-J. Schek (Eds.), Interoperating Geographic Information Systems-Second International Conference, INTEROP'99, Zurich, Switzerland. Lecture Notes in Computer Science *1580, pp. 189-202, Springer-Verlag, Berlin.*

D. Rubine (1991) Specifying Gestures by Example. Computer Graphics *25(4): 329-337.*

R. Scha (1988) Natural Language Interface Systems. in: M. Helander (Ed.) Handbook of Human-Computer Interaction*. pp. 941-956, Elsevier, North-Holland, Amsterdam.*

S. Shapiro and W. Rapaport (1992) The SNePS Family. Computers &Mathematics with Applications *23(2-5): 243-275.*

R. Shariff (1996) Natural Language Spatial Relations: Metric Refinements of Topological Properties. *PhD, University of Maine, Orono, ME.*

B. Shneiderman (1983) Direct Manipulation: A Step Beyond Programming Languages. Computer *16(8): 57-69.*

B. Shneiderman (1990) Future Directions for Human-Computer Interaction. International Journal of Human-Computer Interaction *2(1): 73-90.*

B. Shneiderman (1997) Direct Manipulation for Comprehensible, Predictable, and Controllable User Interfaces. in: International Conference on Intelligent User Interfaces / Proceedings of IUI97*, Orlando, FL, pp. 33-39.*

SIGCHI (1992) Curricula for Human-Computer Interaction. Accessed*: 2/23/1998, http://www.acm.org/sigchi/about.html.*

J. Smith and S.-F. Chang (1996) VisualSEEk: A Fully Automated Content-Based Image Query System. in: ACM Multimedia '96*, Boston, MA, pp. 87-98.*

J. Smith and S.-F. Chang (1997) Querying by Color Regions Using the VisualSEEk Content-Based Visual Query System. in: M. Maybury (Ed.) The International Joint Conference on Artificial Intelligence / Multimedia Information Retrieval*.*

J. Smith and S.-F. Chang (1999) Integrated Spatial and Feature Image Query. Multimedia Systems *7: 129-140.*

R. Srihari and W. Rapaport (1989) Combining Linguistic and Pictorial Information: Using Captions to Interpret Newspaper Photographs. in: D. Kumar (Ed.) Current Trends in SNePS—Semantic Network Processing System, First Annual SNePS Workshop, Buffalo, NY. *437, pp. 85-96, Springer-Verlag, New York, NY.*

M. Stone, K. Fishkin, and E. Bier (1994) The Movable Filter as a User Interface Tool. in: CHI 94*, Boston, MA, pp. 306-312.*

I. Sutherland (1963) SketchPad: A Man-Machine Graphical Communication System. in: Proceedings of the National Computer Conference, AFIPS Spring Joint Computer Conference*, pp. 329-346.*

M. Tate and R. Clelland (1957) Nonparametric and Shortcut Statistics in the Social, Biological, and Medical Sciences*. Interstate Printer and Publishers, Danville, IL.*

S. Timpf (1999) Abstraction, Levels of Detail, and Hierarchies in Map Series. in: C. Freksa and D. Mark (Eds.), Spatial Information Theory-Cognitive and Computational Foundations of Geographic Information Science, COSIT '99, Stade, Germany. Lecture Notes in Computer Science *1661, pp. 125-139, Springer-Verlag, Berlin.*

W. Tobler (1970) A Computer Movie Simulating Urban Growth in the Detroit Region. Economic Geography *46(2): 234-240.*

G. Toussaint (1980) The Relative Neighbourhood Graph of a Finite Planar Set. Pattern Recognition *12: 261-268.*

T. Vo, R. Houghton, J. Yang, U. Bub, U. Meier, A. Waibel, and P. Duchnowski (1995) Multimodal Learning Interfaces. in: ARPA Spoken Language Technology Workshop 1995*, Austin, TX.*

T. Vo and A. Waibel (1993) A Multimodal Human-Computer Interface: Combination of Speech and Gesture Recognition. in: InterCHI'93*, Amsterdam, The Netherlands, pp. 69-70.*

A. Waibel, M. Tue Vo, P. Duchnowski, and S. Manke (1995) Multimodal Interfaces. in: P. McKevitt (Ed.) Artificial Intelligence Review. Special Volume on Integration of Natural Language and Vision Processing *10, pp. 299-319.*

A. Wexelblat (1994) Natural Gesture in Virtual Environments. in: G. Singh, S. Feiner, and D. Thalmann (Eds.), Virtual Reality Software and Technology: Proc. of the VRST'94 Conference. *pp. 5-16, World Scientific, London.*

A. Wexelblat (1995) An Approach to Natural Gesture in Virtual Environments. ACM Transactions on Computer-Human Interaction *2(3): 179-200.*

L. Willauer (1993) Determining GIS Display Characteristics Using Mental Imagery. *M.S. Thesis, University of Maine, Orono.*

C. Williamson and B. Shneiderman (1992) The Dynamic HomeFinder: Evaluating Dynamic Queries in a Real-Estate Information Exploration System. in: Proceedings of the 15th Annual International ACM SIGIR'92 Conference on Research and Development in Information Retrieval*, Copenhagen, Denmark, pp. 338-346.*

P. Wilson (1990) Get Your Desktop Metaphor off my Drafting Table: User Interface Design for Spatial Data Handling. in: Proceedings of the 4th International Symposium on Spatial Data Handling (SDH '90)*, Zurich, Switzerland, pp. 455-462.*

D. Zubin (1989) Natural Language Understanding and Reference Frames. in: D. Mark, A. Frank, M. Egenhofer, S. Freundschuh, M. McGranaghan, and M. White (Eds.), Languages of Spatial Relations: Initiative 2 Specialist Meeting Report. *89-2, pp. 13-16, National Center for Geographic Information and Analysis, Santa Barbara, CA.*

Biography of the Author

Andreas D. Blaser was born in Bern, Switzerland on August 9, 1966. He was raised in Münsingen and Oberwichtrach, both in Canton Bern, Switzerland. Andreas graduated in 1985 from the scientific high school Neufeld in Bern. After serving the mandatory time in the Swiss Army, he continued his education at the Swiss Federal Institute of Technology in Zürich at the Department of Civil and Surveying Engineering in the fall of 1986. During his studies Andreas worked for three years as a System Engineer at IBM in Zürich. After earning his Master's degree at the Swiss Federal Institute of Technology in Zürich (Dipl.-Ing. ETH) in 1992, he worked as a research assistant with the Department of Photogrammetry and Geodesy in the GIS group until the summer of 1995. During this time he supervised M.S. students and was responsible for various projects involving the development of GIS applications.

In the fall of 1995, Andreas entered the Ph.D. program of the Department for Spatial Information Science and Engineering at the University of Maine. He is currently employed as a graduate research assistant with the National Center for Geographic Information and Analysis and will be working for the Environmental Systems Research Institute (ESRI) in Redlands as a GIS Software Development Engineer. Andreas is married and has two children. He is a candidate for the Doctor of Philosophy degree in Spatial Information Science and Engineering from the University of Maine in May, 2000.