8-2000

Image-Based Change Detection Using An Integrated Spatiotemporal Gazetteer

Georgios Mountrakis

IMAGE-BASED CHANGE DETECTION USING AN INTEGRATED SPATIOTEMPORAL GAZETTEER

BY

Georgios Mountrakis

Diploma in Engineering, National Technical University of Athens, Greece, 1998

A THESIS

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

(in Spatial Information Science and Engineering)

The Graduate School

The University of Maine

August, 2000

Advisory Committee:

Peggy Agouris, Assistant Professor of Spatial Information Science and Engineering, Advisor

M. Kate Beard, Associate Professor of Spatial Information Science and Engineering

Max J. Egenhofer, Professor of Spatial Information Science and Engineering

Anthony Stefanidis, Research Assistant Professor of Spatial Information Science and Engineering

IMAGE-BASED CHANGE DETECTION USING AN INTEGRATED SPATIOTEMPORAL GAZETTEER

By Georgios Mountrakis

Thesis Advisor: Dr. Peggy Agouris

An Abstract of the Thesis Presented
in Partial Fulfillment of the Requirements for the
Degree of Master of Science
(in Spatial Information Science and Engineering)
August, 2000

This thesis addresses image-based change detection. Motivation was provided by the lack of algorithms that incorporate in their solution diverse types of pre-existing and complementary information and have the ability to interact with a spatiotemporal environment.

The main differentiation with our approach is that we develop our algorithm within an integrated spatiotemporal environment and we make use of all change evidence that might exist within that environment. In addition, a change resolution model is developed that will distinguish meaningful changes based on user requirements.

A model for change is proposed that establishes a general framework for the incorporation of image analysis techniques. This model is based on the SpatioTemporal Gazetteer. A theoretical analysis of change modeling resulted in the Differential *Spatiotemporal Gazetteer*. This modified version of the original gazetteer is composed conceptually of a Geographic Identity Information Level, a Change Indexing Information

level and a collection of child sources at the Child Information Level. Dependencies between these components are presented. In addition to that, two new components are introduced, the Change Detection Tools and the Knowledge Base expanding the gazetteer to a *broader spatiotemporal environment.* Information flow within the environment is described and component interaction is presented, focusing on the change detection aspects.

In spatiotemporal applications, meaningful changes vary according to object type, level of detail, and nature of application. To compensate for this, we introduce two user-defined functions, the Minimum Spatial Element and the Minimum Temporal Element. These functions act as thresholds in the spatial and temporal domain, respectively, and allow the user to establish a resolutional framework in the spatiotemporal domain.

With the proposed spatiotemporal model, we improve the change detection process, by providing validity of the datasets used, accuracy during the detection process, and a framework for storing the obtained results. The digital image analysis method that was developed automatically identifies object outline changes in sequences of digital images. This change detection technique is based on least squares template matching (LSM). We extend LSM to function in a differential mode. In doing so, we integrate object extraction from digital imagery with change detection in a single process. By using image orientation parameters and positional data we can reduce the problem of 3-D object monitoring to an image-space 2-D matching problem.

In this hybrid approach, area- and feature-based matching are combined in one step, since raster and vector datasets are integrated to enhance our solution. Analysis of the edge geometry within a template, before matching takes place, improves the accuracy and reliability of the presented technique based on response time and obtained results. As a post-process, actual change is distinguished from different representations of the same object due to sensor inaccuracies, through fitting models of known systematic photogrammetric errors in the exterior and interior orientation process. The obtained results update the corresponding components of our spatiotemporal model through the detected change information. In doing so we establish a spatiotemporal model that makes use of the change detection results.

Acknowledgments

This work was partially supported by the National Imagery and Mapping Agency through grant number NMA202-98-1-1113 for the National Center for Geographic Information and Analysis, and by the Evgenidio Foundation through a graduate scholarship.

For their support, collaboration and assistance with my thesis, I would like to thank my advisor Peggy Agouris, my research co-advisor Anthony Stefanidis, and the reviewing members Kate Beard and Max Egenhofer.

Special thanks go to my family for their support and guidance. Also I would like to thank whoever contributed during the development of this work, intentionally or not.

Finally I am grateful to my previous advisor, Giorgos Karras, for showing me the right way for my graduate studies, and Gillian Avruskin for her patience and encouragement.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

One of the major tools used to model accurately the real world is a geographic information system (GIS). In such a system and in order to capture the dynamic world, a GIS approach that handles change information should be employed. Such a model is useful in many application domains: historical geography, meteorology, physical geography, and engineering fields like city planning, landscape architecture, cadastral systems and transportation. Scientists have always been eager to understand the past and the present, and to predict the future. In order to achieve this, the ability to perform a dynamic analysis of information over time is crucial for GIS applications.

Current geographic information systems tend to follow an inherently static approach to geospatial information management. Small amounts of information are typically synthesized into map-like data snapshots. Many times in the mapping process, various sources of information containing multiple representations of features are reduced to a single representation (Langran, 1993). This static approach leaves large amounts of collected data unused and offers limited communication capabilities. Accordingly, this approach is unsuitable for today's applications, where the need for dynamic geospatial information is increasing. Change, whether it is the very dynamic change of mobile objects, periodic fluctuations of fixed objects, abrupt change such as that resulting from catastrophic events, or more routine incremental change, is valuable information to which an information system should provide direct access. Information on change is important to better understand why certain conditions exist – either now or in the past – and to make predictions of what the conditions or configurations of an entity will be at some future time.

Extracting change within integrated geospatial environments containing digital imagery is a challenge of dual nature. It comprises automatic change detection (Huang et al.,1994), and the fundamental issue of modeling/representing change (Egenhofer and Golledge, 1998). A rather surprising tendency in object extraction algorithm development within the photogrammetric and computer vision communities is that the majority of approaches treat geospatial information collection as a static process (Gruen et al, 1995; Gruen et al, 1997). They formulate the problem as extracting an outline from an image without consideration of prior information that might exist for the depicted area in general, and the specific object in particular. Considering today's practice trends, this static view tends to become increasingly obsolete. Instead, object extraction becomes a crucial part of larger cycles of GIS updates, and information exchange is presented as essential.

1.1 Problem Statement

Understanding how objects behave and change over time is a fundamental issue that has to be addressed in order to model the real world in an appropriate manner. Change needs to be formalized in order to create an effective spatio-temporal environment (Galton, 1993). An emphasis on the temporal aspect of data should be given for managing the evolution of objects through time by modeling and detecting change in digital imagery.

In order to achieve this modeling and detection of change we should first consider how evidences of change influence this approach. Major evolutions in sensor/scanner technology have resulted in the availability of constantly increasing volumes of digital imagery. The geosciences and spatial information engineering have been greatly affected by this development. Parallel to this development, digital photogrammetric applications have become more robust with the development of novel digital image analysis methods and the production of robust software tools. Accordingly, we are moving from the experimental use of few images to large-scale projects. Nowadays, one could claim that digital images have practically replaced analog ones (e.g., prints, diapositives, or negatives) as the popular medium for the extraction of precise and up-to-date spatial information such as digital elevation models or features with their 3-D coordinates. The increased volumes of digital imagery, in conjunction with the evolutions in data collection and processing have drastically increased the amount and type of the available geospatial information. Modern environments are now characterized by higher integration and should be able to handle multi-resolution, multi-type and multi-temporal information. Moreover, this aggregation of information is altering the way geospatial analysis is performed, by providing us with substantial prior knowledge for the behavior of objects through time (e.g., make use of information showing the ability of an object to move). The development of novel models to capture, categorize and store geospatial object change, and to make them available in an efficient way for querying and further detection/prediction of change, is becoming increasingly critical.

Conceptual models of time (Allen, 1983) and their applications on spatio-temporal environments (Yuan, 1994) have been introduced in the past, but the lack of a widely accepted approach that would model change both at the spatial as well as the temporal domain is still notable. Conventional GIS data models still emphasize static representations of reality. Geographic information for a given area is decomposed into a set of single-theme layers as regular (raster) or irregular (vector) tessellation models (Frank and Mark, 1991). These layers limit our capabilities to represent dynamic information such as transitions and motion within a GIS. The lack of data representation schemata to integrate GIS data with models for spatiotemporal processes is a major shortcoming in current GISs.

In our approach we present a framework for the integration of digital images and complementary GIS datasets in a model that provides explicit information about spatiotemporal change. We make use of the SpatioTemporal Gazetteer (Beard and Agouris, 1997) as a model that handles more effective multiple information resources than the traditional map model. We extend this model to facilitate the image-based change detection proposed in this thesis and to allow it to function in a differential mode through a change-oriented approach. The components of the proposed model are introduced and we discuss information flow in such an environment.

A significant functionality of a spatiotemporal model is the ability to express explicitly changes that may be found implicitly in various sources and object representations. Defining a common measurement framework to express spatiotemporal change is a challenging task, as meaningful changes vary according to level of detail, nature of application, and type of object (Mountrakis et al, 2000). Furthermore, the context of change, as determined by applications and user needs in queries, can be considered to be spatial in nature, temporal in design, or a combination of both. The communication of change requires the consideration of users who have diverse needs, and various levels of understanding of the content and organizational schemata of the databases that they query. Related work has been performed in identifying and formalizing operations in the temporal domain in order to support the integration of different databases (Bettini et al., 1997) and also on change granularity issues (Montanari et al., 1992; Wang et al., 1993; Hornsby and Egenhofer, 1999). With this approach we extend the concept of change granularity in the design and query process of a spatiotemporal model.

In signal processing and image understanding, relevant work has been performed on man-made object extraction (Gruen et al., 1995; Gruen et al., 1997) and in human motion understanding (Black et al., 1996; Yacoob and Black, 1999). The approach presented in this thesis is based on an extension of least squares matching (LSM). In the past LSM was extended to support matching of 3-D datasets (Gruen et al., 1994) and image queries (Agouris and Stefanidis, 1998). In this approach the adjustment method is modified to detect boundary change within an integrated environment. The established mathematical foundation of LSM is used for statistical analysis of the obtained results and realistic evaluation of the algorithm's performance. Interaction of the algorithm with a knowledge base to provide reasoning in our process is also incorporated to take full advantage of prior information that might exist.

1.2 Research Questions and Objectives

The goal of this work is to enhance operations that extract change information from multimedia information sources, and specifically from images. While trying to find an answer to this issue, many others arise as a logical sequence.

First a spatiotemporal environment should be developed to handle the primitives of image analysis techniques. In addition, this environment should have the ability to store and retrieve the results obtained in an efficient way. An applicable database model should be developed to interact with the algorithm both at the input as well as the output level.

Another open question is how the user would specify the change resolution that is of interest for the specific application or needs. This work incorporates this flexibility by providing a multi-resolutional query method and its translation to the algorithms datasets.

The described in this thesis image analysis approach is based on the least squares matching technique. Least squares matching is a single-process technique. The design of this method does not support change detection. To compensate for this, a modified version of LSM is developed that is designed to function in a differential mode. This approach is extended to a universal one so that change detection is not an isolated procedure, but rather interacts within a spatiotemporal environment. Furthermore the appropriate adjustments should be performed to optimize prior knowledge incorporation in the detection of change.

Accordingly with the above, the hypothesis of this thesis is:

“the proposed universal differential least squares approach outperforms the traditional single-process least squares matching in change detection on digital imagery.”

1.3 Motivation and Applications

The major motivation for this research is the lack of a theoretical framework that would have the ability to model, query and detect change within one integrated environment containing digital imagery. The interactions and dependencies that these processes introduce were considered during the design of each component (model, query, and detection). This more global approach overcomes issues of incompatibility and miscommunication.

Part of the motivation in designing the proposed model was the need to support different resolutions of change. Furthermore, the way queries and change detection process are formulated allow the integration of multiple users with diverse needs (e.g., ranging from a local township GIS office to national emergency response crews). Filtering of the obtained results guides the query in the desired spatial and temporal resolution, providing a major advantage that was not addressed in the literature before. The structure of the model establishes the framework within which change is categorized, analyzed and stored. Previous work on this topic has been performed in the GIS (Armstrong, 1988; Yuan, 1994; Langran and Chrisman, 1988; Worboys, 1992; Raper and Livingstone, 1995; Hazelton, 1998) and computer science (Allen, 1983; Freksa, 1992; Segev and Shoshani, 1993; Tryfona and Jensen, 1999; Sellis, 1999) communities.

In image processing, no significant work has been done in detecting change through different representations by using multi-type, multi-temporal data in conjunction with prior knowledge for the object under examination. Most of the approaches are limited in determining pixel intensity changes, and a few of them that go beyond that do not take full advantage of the preexisting information (Huang et al., 1994; Huertas and Nevatia, 1996; Chellappa et al., 1993). Motivation for this part of the research was provided by the lack of algorithms that would incorporate in their solution multiple types of pre-existing (raster, vector, database, etc.) information, and have the ability to interact with a spatiotemporal environment, both at the input as the output level.

The intended audience for this specific research comprises experts from a variety of related fields of study, such as remote sensing, digital photogrammetry, computer science, GIS and artificial intelligence. Our approach to managing geospatial information is of interest to a large variety of systems such as geographic information systems, environmental information systems, remote sensing systems, etc. Spatiotemporal applications, such as fleet management, air traffic control, global change (as in climate or land cover changes), transportation (traffic surveillance data, intelligent transportation systems), social (demographic, health, etc.), multimedia (animated movies), geological, archaeological and monitoring applications also could benefit from this approach.

1.4 Major Contribution of Thesis

The contribution of this research work is directly related with the previously defined objectives. At a more general level, a major contribution is the establishment of a spatiotemporal environment that allows the integration of multi-temporal, multi-resolution, multi-type information for image-based change detection. This environment supports information flow between three major operations: modeling, querying and detection of change.

Within this environment, the structure of the model provides the essential mechanism for storing, representing and retrieving the detected change. A change resolution filtering based on the user needs is introduced. The combination of these two contributes to the algorithm by providing and updating the appropriate datasets involved in the change detection. These datasets are constrained in the desired change resolution providing no redundant information to, and application of, the change detection process. Another component of the model shows the ability of an object to change. We make use of this information by limiting the search window based on expected behavior.

Our novel digital image analysis approach provides a tool to detect changes in building outlines by combining prior knowledge and object extraction in a single process. At the input level, multi-type (e.g., raster, vector, database) spatiotemporal information is analyzed to improve the reliability of the matching technique. During the matching procedure, the statistical origin of least squares contributes to a robust evaluation of the performance. By applying reasoning to the obtained results, object changes are distinguished from misregistration errors through interpolation of well-established photogrammetric models (e.g., interior and exterior orientation models). If such errors are detected, the update of the corresponding metadata of the source contributes to a more reliable future use of that image.

1.5 Thesis Organization

The remainder of this thesis is organized into five chapters. Chapter 2 provides an overview of the current research status in change detection and modeling in spatiotemporal environments. A detailed description of the components of the spatiotemporal model, used in this thesis, follows and the interactions between them are presented (Chapter 3). Chapter 4 is dedicated to the theoretical issues of the image analysis technique, developed to automatically detect changes in building outlines. The implementation and evaluation of the change detection approach are discussed in Chapter 5. Finally the conclusions summarize and evaluate the major findings of this thesis and provide ideas for future developments (Chapter 6).

Chapter 2

Background and Relevant Work

The purpose of this chapter is to provide an overview of recent and current research relevant to this thesis. Since several issues addressed require a combination of knowledge from different areas of study, such as computer science, photogrammetry, remote sensing, GIS and artificial intelligence, the organization of this chapter follows an issue-based rather than a discipline-based approach. Accordingly, this overview is arranged into two sections. The first one describes existing work on modeling and querying in the spatiotemporal domain. The second section presents the general framework and the theoretical background of the digital image analysis technique that was developed in this thesis.

2.1 Strategies for Modeling and Querying Spatiotemporal Data

In this section an overview of the relevant to this thesis work in the spatiotemporal modeling and querying domain is presented. The structure of the literature analysis follows the sequence of

- a general description of database models,
- the contributions in the spatial domain,
- relevant work from the temporal domain and

- issues addressed in the spatiotemporal domain.

The last three sections are further arranged based on related work performed in modeling, queries and granularity issues in each domain separately.

2.1.1 General Database Models

Substantial work has been presented in the database field, resulting in the proposal of several models. In general these models are classified in two categories: the relational and the object models.

The *relational* model was introduced by Codd (1970) and has been the inspiration for an entire generation of database management systems. These systems are based on the concept of a relation, which is a set of tuples. A tuple is a set of facts that are related to each other in some way. Each fact in a tuple is a datum whose values are typically implemented through a specified domain (e.g., the domain of all integers, the domain of all character strings of length 255 or less). A relation is a (possibly complete) subset of all the possible tuples formed by the Cartesian product of the domains. Since tuples are sets (of values) and a relation is also a set (of tuples), relations are sets of sets.

The solid theoretical underpinning, inherited from the well-established mathematical data model, allows ad hoc query languages whose queries can be automatically compiled, executed, and optimized without resorting to programming. The semantics of the relational algebra are sound and complete, enabling users to easily anticipate the result of a given query. Also, the relational model's integrity constraints are very helpful in ensuring that structural changes do not adversely effect the meaning of the database.

The object model was initially introduced as an extension of object-oriented programming concepts (Albano et al., 1986; Lécluse and Richard, 1989; Carey et al., 1988). Objects are distinct and are *conceptually* collected together into meaningful groups. These groups are called *classes.* An object grouping is meaningful, because objects of the same class have common behaviors, attributes, and relationships with other objects.

The object model provides the following advantages, derived from object-oriented programming:

- great freedom regarding complex data structures,
- inheritance, meaning that a hierarchy among classes can be created and attributes can propagate from superclasses to subclasses, and
- encapsulation, allowing the implementation of any behavior to be changed without effecting any other class in the system. This helps uncouple the classes and reduces the complexity of the system.

2.1.2 Spatial Domain

2.1.2.1 Modeling Space

Currently two broad and opposing conceptual approaches can be identified that are modeling space within GISs (Chrisman, 1975; Chrisman, 1978; Peuquet, 1984; Smith and Mark, 1998). The *field-based* model treats spatial information as a collection of spatial functions transforming a space partition to an attribute domain, creating a collection of fields. Each field defines the spatial variation of an attribute as a function from the set of locations (Worboys, 1995). The field-based approach conceptualizes the relation as divided into variations of single or multiple attributes (columns). The functions used try to relate the spatial framework to the other attributes. The table 2.1 shows a field-based representation of an attribute (temperature) in space.

Code	X	Y	Temperature
001901	1367.22	1255.62	55.9
001902	1502.87	1158.37	57.1
…	…	…	…
001934	1345.32	1409.44	56.7

Table 2.1: Example of Field-based Representation

Another way to treat conceptualization of space is through an *object-bused* approach. In this case the functions try to relate the attributes of the entities to a spatial framework. It treats the information space as if it is populated by discrete, identifiable, spatially referenced entities (Shekhar et al., 1999). For each object a set of attributes is assigned, so a GIS application is mainly populated by a collection of spatially referenced objects. These spatially referenced objects may have multiple spatial objects as attributes (e.g., a house may reference a polygon as a boundary and a point as its seed). **As** an example, in an object-based approach space would be expressed through an object, namely Boardman Hall, as shown below.

$Object.\ Code = 00$ 1999

$Object.\ Type =$ Building

$Object.Name =$ Boardman Hall

$Object.X = 1345.32$

$Object.Y = 1409.44$

$Object.Z = 32.44$

$Object.Area = 450.72$

2.1.2.2 Spatial Queries

In order to support the spatial component within the structure of a traditional database management system (DBMS), two different architectures can be identified, based on their approach in integrating spatial and non-spatial data.

The first one is the *dual architecture spatial database* that extends DBMSs by adding a parallel spatial subsystem to store and manipulate spatial data. Object attributes are decomposed in spatial and non-spatial, and are stored separately. Examples of dual architecture systems are the GEOgraphic Query Language (GEOQL) (Ooi and Davis, 1989) and the Spatial And Non-spatial Database system (SAND) (Esperanca and Samet, 1997). Other approaches include Query-by-Pictorial-Example (Chang and Fu, 1980) and a Quel extension for image processing (Embey and Nagy, 1986). As described in Egenhofer (1992), query languages that treat space as an extension find it difficult to incorporate and express spatial complexity in their models.

Extensive spatial database architectures are another way to incorporate support for spatial operations. In this approach, operations that manipulate spatial data types (SDT) are added directly to the DBMS kernel. This allows customization in the structure and the behavior of SDT. In addition to that, SDTs are treated at the same level as traditional field types (e.g., integer, string). As an example we could mention GRAL, where relational algebra is used as its query language, query plans and optimization rules (Becker and Guting, 1992).

2.1.2.3 Spatial Granularity

An inherent property of objects in the world is that they can only be perceived as meaningful entities over certain ranges of space scales. Therefore if one aims at describing the structure of unknown real-world signals, then a multi-scale representation of data is of primary importance.

Scale is neither a new issue, nor is restricted to geographic information scientists (Goodchild and Proctor, 1997). Scale variations have long been known to constrain the detail with which information can be observed, represented, analyzed, and communicated. Changing the scale of data without first understanding the effects of such action can result in the representation of processes or patterns that are different from those intended. The spatial scaling problem presents one of major impediments, both conceptually and methodologically, to advancing all sciences that use geographic information. In today's information era, massive amounts of geographic data are collected from various sources, often at different scales. Before these datasets can be used for problem solving, fundamental issues describing this multiscale integration must be addressed.

Issues of scale affect nearly every GIS application and involve questions of scale cognition, the scale or range of scales over which phenomena can be recognized, optimal digital representations, technology and methodology of data observation, generalization, and information communication. These are very different types of questions. Effective research in the area of scale requires interdisciplinary efforts of geographers, spatial geostatisticians, cartographers, remote sensing specialists, domain experts, cognitive scientists, and computer scientists. Recent attention is focused on formalizing the concept of scale, on developing theory, and on exploring robust methods for information representation, analysis, and communication across multiple scales. Research on scale has been presented in geography (Hudson, 1992), remote sensing applications of GIS (Quattrochi and Goodchild, 1997), cartography (Buttenfield and McMaster, 1991), spatial statistics (Wong and Amrhein, 1996), hydrology (Sivapalan and Kalma, 1995), and ecology (Ehleringer and Field, 1993) among other areas.

In the vector domain several attempts can be sited in generalizing shapes (de Berg et al., 1997) and digital elevation models (Floriani et al., 1996). In the raster domain and in particular the computer vision community has developed a special type of multi-scale representation, the *scale-space representation* (Witkin et al, 1986) in order to handle image structures at different scales in a consistent manner. The basic idea is to embed the original signal into a one (uniform scale along X and Y directions) or two (separate scale factor on X and Y directions respectively) parameter families of gradually smoothed signals. With this approach the fine scale details are successively suppressed.

Scale-space concepts were introduced in the seventies in the form of hierarchical information structures (Tanimoto and Pavlidis, 1975). The first formal representation and development is credited to Witkin (1983). Following that, a number of approaches to multi-scale representations have been developed, which are more or less related to scale-space theory, notably the theories of pyramids (Burt, 1984; Meer et al., 1987), wavelets (Mallat, 1989) and multi-grid methods (Lindeberg, 1994). Despite their qualitative differences, the increasing popularity of each of these approaches indicates that the crucial notion of scale is increasingly appreciated by the computer vision community and by researchers in other related fields, such as remote sensing and GIS.

2.1.3 Temporal Domain

2.1.3.1 Modeling Time

Time is a critical aspect when modeling changes. This brief literature review of temporal modeling begins with a presentation of how time is conceived and extends to existing temporal database models.

Several models of time have been adopted to represent temporal information in an efficient way. In the *linear* model, time advances step by step from the past to the future. The *cyclic* model is used for recurrent processes like seasons, months and weeks. In the *branching* model several parallel time lines exist, representing dependencies between entities. A time model can be *discrete* or *continuous.* Discrete models adopt a structure of points in time that are isomorphic to the natural numbers, so each point has a single successor. In the case of continuous model, the time structure is isomorphic to the real numbers so no gaps exist. A more thorough analysis of time models can be found in Frank (1998).

When a time value is associated to an object, a *time-stamp* is assigned. Time-stamps can either be discrete, corresponding to *time instants* (Montanari and Percini, 1993), or continuous creating *time intervals* (Allen, 1983). Allen's work has provided some advantages, mostly concentrated on temporal reasoning, by allowing insertion of uncertainty and flexibility in the events order.

In current temporal systems, two time dimensions are of general interest in the context of databases, the *transaction* and the *valid* time (Snodgrass, 1992). The valid time of a fact is the time when the fact is true in the modeled reality and is typically supplied by the user. The transaction time is the time that the fact is recorded in the database. Transaction times are system generated and monotonically increasing.

A temporal database is categorized as transaction-time, valid-time or bitemporal (Kumar et al, 1998), according to which temporal dimension(s) it supports. In a more formal way, database models may be classified according to their consideration of time as follows (Snodgrass, 1992):

1. *Static databases.* This case represents most of the traditional databases. No support for time is incorporated, and only the latest version of data is recorded, without any ability to reconstruct temporal sequences.

2. *Static rollback databases.* In this type of database, transaction time is recorded in an indexing form, but the user is not allowed to make any modifications to the previous states.

3. *Historical databases.* These databases provide the ability to capture the valid time of the events recorded. Users can navigate through different time instances, but no modifications can be made.

4. *Bitemporal databases.* Clearly both transaction and valid time are needed in order to model reality. This is the most recent and successful type of temporal database. Both transaction and valid time are captured (Snodgrass and Ahn, 1985; Armstrong, 1988; Al-Taha and Barrera, 1990) and both system and systems content time changes are tracked (Snodgrass and Ahn, 1985; Snodgrass, 1992).

For a more detailed overview of existing approaches the reader is referred to Tansel et al., (1993) Tsotras and Kumar, (1996) Ozsoyoglu and Snodgrass, (1995).

2.1.3.2 Temporal Queries

Optimizing temporal queries is a much harder task than that of conventional queries. First the relations that the time component introduces add significant complexity to the formal representation of such a language. Second the predicates used in temporal queries are harder to optimize. In traditional database applications, predicates usually have an equality form (e.g., name = Boardman Hall). In temporal queries the case of inequality predicates (e.g., time >11/2/1998) appear more often, thus making the internal structure a more involved process.

On the other hand using time in a query process offers a significant advantage in optimization and indexing. Time advances in one direction, allowing natural clustering and sorting order. Also, many relations are contiguous in time, for example one starting directly after the other.

Several languages have been proposed for querying temporal databases, like TQUEL (Snodgrass, 1987), TOSQL (Ariav, 1986) and HSQL (Sarda, 1990). Lately, TSQL2 has combined many years of research results into a single, comprehensive language (Snodgrass et al., 1995). Some of its components are incorporated in a part of SQL3, called SQL/Temporal.

A comprehensive overview of temporal indexing strategies may be found in Salzberg and Tsotras (1999). Some of these indexes are based on B^+ trees, performing indexing of values on a single key, while some others try to index ranges of multiple keys, by following an R tree approach.

2.1.3.3 Temporal Granularity

The inclusion of multi-temporal resolution (also defined as temporal granularity) in temporal databases is the focus of many researchers in the active field of temporal databases. They are mostly driven by implementation issues, and the majority of the approaches consider only temporal data, not spatiotemporal as it is later analyzed in section 2.1.4.3.

We begin our overview with Anderson's work, since he is considered to be a pioneer in the research for modeling time resolution (Anderson, 1982). Anderson pointed out the need to model times at multiple granularities. Clifford and Rao further developed Anderson's framework by adding a granularity chain (a complete ordering of granularities) and finer granularity conversions between times (Clifford and Rao, 1987).

Wiederhold et al. made Clifford's and Rao's theoretical work more concrete by proposing specific semantics for temporal comparisons at mixed granularities (Wiederhold et al., 1991). Wang et al. generalized the granularity chain to a lattice and proposed semantics for moving times up and down the lattice (Wang et al., 1993). They also proposed temporal modules and extended temporal modules that provide access to temporal relations via windowing functions, each in terms of a different time unit.

Barbic and Pernici discuss relative, absolute, periodic, and imprecise times at different Gregorian granularities for office information systems in the context of constraint triggers (Barbic and Pemici, 1985). Montanari et al. investigated a slightly different problem, that of extending the granularity chain to cover macro-events, i.e., events with duration (Montanari et al., 1992).

Relevant work on time granularities has been performed also in other areas such as logic programming (Montanari et al., 1992) and real time system specification (Ciapessoni et al., 1993). In these papers the emphasis is on embedding temporal granularity notions into a logical formalism. Finally, several papers address the representation and implementation of calendars and granularities (Chandra et al., 1994; Leban et al., 1986; Niezette and Stevenne, 1992). The set of granularities expressible in these proposals is often characterized only by a representation language.

2.1.4 Spatiotemporal Domain

2.1.4.1 Spatiotemporal Modeling

Spatiotemporal applications deal with time-varying spatial data. These two components have mostly been treated as independent domains of research, and a simple combination of spatial and temporal theories is not adequate to embrace all the new issues raised by their integration. The main reasons lie in the complexity of their components: space by itself is a complex and intricate issue, involving, among others, position of objects, and spatial attributes that change values depending on specific locations (Peuquet and Wentz, 1994). On the other hand, research on temporal issues is related mainly to physical design aspects and improvements of systems performance (Ozsoyoglu and Snodgrass, 1995).

Several methods have been introduced that deal with the combined effort of spatial and temporal information. Story and Worboys (1995) proposed a design support environment for spatiotemporal databases, focusing on the integration of time application data. Allen et al. (1995) presented a generic model consisting of objects, states, events and conditions for explicitly representing casual links within a spatiotemporal GIS.

Worboys (1994) proposes a unified model for information which is referenced to two spatial dimensions and two temporal dimensions (database and event times). Claramunt et al. (1997) present a set of design patterns for spatiotemporal processes expressed in an object-relational data model. Hazelton (1998) discusses some of the operational requirements of a multi-temporal 4-D GIS, and describes the role of the data structure and software architecture in realizing these requirements. Often domain experts give their own solutions and modeling techniques to these issues, mainly driven by the need for fast and applicable answers. The result is spatiotemporal systems tightly coupled with specific software and hardware (Batty and Xie, 1994a; Batty and Xie, 1994b; Yuan, 1994).

A very promising effort began in Europe in 1996 with the Chorochronos project (Chorochronos, 2000). A variety of papers address spatiotemporal modeling, fuzziness and accuracy, querying methods spatial and temporal resolution and other relevant topics. In particular for spatiotemporal modeling a recent summary of this work can be found in Tryfona and Jensen, (1999) and Sellis, (1999).

Outside the GIS community, spatiotemporal models are emerging as a very important topic in several other disciplines including neurobiology and artificial neural networks, optical engineering and physics. Many fundamental problems exist in this area. Examples include understanding the capabilities of nonlinear dynamical systems on a lattice and of networks of spiking neurons, both natural and artificial (Silva et al., 1997), motion estimation through energy models (Adelson and Bergan, 1985), and statistical descriptions of degrees of freedom in spatiotemporal chaos in the field of theoretical physics (Cross and Tu, 1994).

2.1.4.2 Spatiotemporal Queries

A review of the related literature, indicates a lack of a unifying approach to spatiotemporal queries. A general notation for spatiotemporal queries was first attempted by Tsotras et al. (1998). This notation associates each portion of the selection criteria with one or more explicit attribute values, spatial dimensions, or temporal dimensions to a descriptive entry. Query qualifications included point containment and interval intersection.

Following the same distinction of 1D and 2D dimensions, *moving points* and *moving regions* are considered basic abstractions in Erwig et al. (1999). In their research moving points and moving regions are viewed as three-dimensional (2D space + time) or higher-dimensional entities whose structure and behavior is captured by modeling them as abstract data types. They propose a query method based on the above model.

A more detailed description, including implementation issues, is provided in Erwig and Schneider (1999). In particular, they demonstrated how SQL can be extended to enable spatiotemporal querying. They called this extended query language STQL (Spatio-Temporal Query Language).

2.1.4.3 Spatiotemporal Granularity

In Shahar and Molina (1998) a theoretical framework and a case study was presented for reusing the same conceptual and computational methodology for both temporal abstraction and linear (unidimensional) space abstraction. The method used, known as knowledge-based temporal abstraction, abstracts high-level concepts and patterns from time-stamped raw data using a formal theory of domain-specific temporal-abstraction knowledge.

In Hornsby, (1999) and Hornsby and Egenhofer (1999) granularity is decomposed into granularity of *objects* and *transitions.* This approach identifies the basics for extending granularity concepts to the communication of a model, but is limited to qualitative aspects of the spatiotemporal domain. Related work on the qualitative spatiotemporal level is also provided in Euzenat, (1995).

Spatiotemporal granularity is also used in the image processing field. As an example the work in Dumas et al., (2000) exploits the concept of granularity to design a video metadata model that addresses both logical structuration and content annotation in an orthogonal way.

2.2 Image-Based Change Detection Techniques

Image analysis is commonly faced with spatiotemporal change detection mostly within the context of moving objects (Aggarwal and Nandhakumar, 1988; Irani and Anandan, 1997; Yip and Pong, 1996; Zheng and Chellappa, 1995). The majority of the algorithms that fall in this category work with raw sequences of images (i.e., video) without incorporating prior knowledge in their analysis and without any interaction with a broader spatiotemporal information environment.

Among the early attempts in detecting change through different representations we recognize Lillestrand, whose approach was based on determining pixel intensity changes (Lillestrand, 1972). Other notable examples may be found in Huang et al., (1994) Huertas and Nevatia (1996) and Chellappa et al. (1993).

The approach introduced in this thesis is based on a modification of the image matching problem. Therefore a literature review of past image-matching techniques is provided below.

2.2.1 Image Matching Techniques

In computer vision, photogrammetry and remote sensing, matching can be defined as the establishment of correspondences among two or more raster datasets. These datasets are mostly digital images, but often may be maps, object models or other types of GIS data. Many steps of the photogrammetric process are linked to matching in one way or another such as:

- interior orientation: the image of a fiducial is matched with a two-dimensional model of the fiducial,
- relative orientation and point transfer in aerial triangulation: parts of one image are matched with parts of other images in order to generate tie points,
- absolute orientation: parts of the image are matched with a description of control features (e.g. ground control points),
- generation of digital terrain models (DTM): parts of an image are matched with parts of another image in order to generate three-dimensional object points,
- image interpretation: parts of the image are matched with object models in order to identify and localize the depicted scene objects.

Image matching involves the comparison of two dimensional representations of three-dimensional objects. This projection of a 3-D space onto 2-D datasets induces certain loss of information. This is most evident in the case of occlusions, where an object may be partially hidden due to the configuration of other objects at the capturing scene and the specific sensor orientation. Image matching belongs to the class of so called inverse problems, which are known to be ill-posed. A problem is ill-posed, if no guarantee can be given that a solution exists, is unique, and is stable with respect to small variations of the input data. Image matching is considered an ill-posed problem, because for a given point in one image, a corresponding point may not exist due to occlusion, there may be more than one possible match due to repetitive patterns or a semi-transparent object surface, and the solution may be unstable with respect to noise due to poor texture.

In order to find the solution of an ill-posed problem one usually has to employ an optimization function that may exhibit many local extrema. This limits the pull-in range of matching processes and requires the availability of precise a priori approximations of the conjugate point locations. Other stringent requirements may also exist in determining initial values for unknown parameters (e.g. approximations of rotations and/or scale between two photos). Moreover, usually there is a large search space for these parameters, and numerical instabilities may arise during the computations.

Ill-posed problems can be converted to well-posed problems by making use of additional knowledge. Fortunately, a wide range of assumptions usually holds true when dealing with photogrammetric imagery:

- the gray values of the various images have been acquired using one and the same or at least similar spectral band(s),
- illumination and relevant atmospheric effects are constant during image acquisition,

- the object surface is piecewise smooth and opaque, and exhibits a more or less diffuse reflection function,
- coarse initial values such as the approximate overlap between the images or an average object height are commonly known.

Depending on the actual problem at hand additional assumptions may be introduced, or some points of the above list may be violated. It is this mixture of necessary assumptions, which makes the design of universal image matching algorithms difficult, and has lead to the development of different algorithms in the past. Most matching algorithms proposed in the literature contain implicitly or explicitly a combination of assumptions about the depicted scene and the image acquisition conditions.

Rather than trying to describe these algorithms as a whole it seems more appropriate to categorize them based on the primitives that are selected for matching. These primitives define the models used for geometric and radiometric mapping, the way the optimal match is computed and the strategy employed to control the matching algorithm.

2.2.2 Types of Matching Primitives

The distinction between different matching primitives is probably the most prominent characteristic distinguishing various matching algorithms. One of the reasons is that this selection influences the whole development of such an algorithm. The primitives fall into two broad categories: gray value windows and image features extracted a priori. The resulting algorithms are usually termed *area-based matching*

(ABM), and feature-based' *matching* (FBM), respectively. Note that when talking about ABM or FBM not only the selection of the primitives, but the whole matching process are implied. Hybrid techniques also exist, combining the strengths of the first two methods (Doorn et al., 1990). In any case there is a choice between local and global support for the primitives. The terms *local* and *global* are not sharply defined. Local refers to an area seldom larger than about 50 * 50 pixels in image space, global means a larger area and can comprise the whole image.

2.2.2.1 Area-based Matching

In area-based matching small windows composed of gray values serve as matching primitives. The approach involves the extraction and precise matching of conjugate pixel windows or patches (Fig. 2.1). Here we should note that the terms "image window" and 'Image patch" are synonymous in this context and are used interchangeably in this thesis. These patches are in the form of two-dimensional arrays of gray scale values. The gray values are regarded as samples of the continuous image brightness function in image space, and concepts of signal processing can be employed for further computations. Best match between these two patches is where their collective

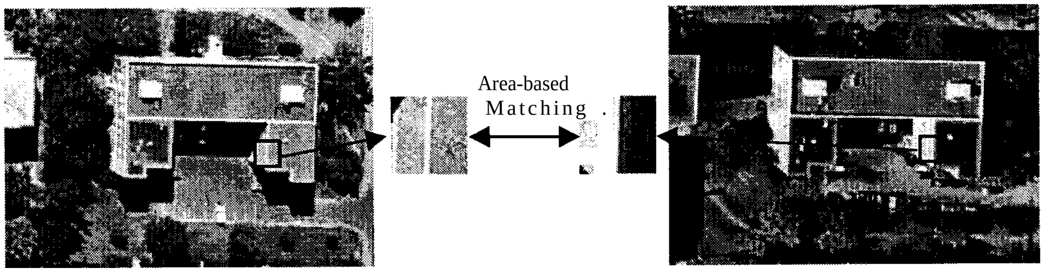


Figure 2.1: Visualization of Area-based Matching

difference in gray values is minimized (Ackerman, 1984). The windows can be extracted very fast, and the actual matching methods are rather straightforward. Also, ABM has a high accuracy potential in well-textured image regions, and in some cases the resulting accuracy can be quantified in terms of metric units.

A disadvantage of this approach is the sensitivity on actual gray values. The same object might have different radiometric and geometric representations in two different images. Such deviations can be caused by illumination and image perspective changes. Blunders can occur in areas of occlusions, and poor or repetitive texture. To correlate these different representations of the same object in two different patches, models expressing geometric and radiometric transformations are used. Typically, a two-dimensional affme transformation is used to describe the geometric relationship combined with added parameters to eliminate the radiometric differences.

Another limitation of area-based matching is that it requires good approximations for the initial patch locations (Baltsavias, 199 1). The 'hull-in" range of this approach is defined as the maximum distance between the initial approximate position of a patch and its true conjugate position that can still be bridged and provide a correct match.

If ABM is applied locally, several techniques can be chosen, two of which are the cross correlation method and the least squares matching approach (Fbrstner, 1982; Gruen and Baltsavias, 1987). ABM can also be carried out globally using connected windows (Rosenholm, 1987). In this case poor and repetitive texture can be successfully addressed up to a certain extent.

2.2.2.2 Feature-based Matching

In FBM, features are extracted from each image individually prior to matching them. It is a model based on a matching strategy that proceeds from *coarse* to fine (Greenfeld, 1987). The feature extraction can be done in a number of ways, depending on the matching requirements (e.g. Förstner operator, edge detection filters). The result of this extraction is a list containing the features and their descriptions for each image. Only these lists are processed further. It should be noted that the features are discrete functions of position: after feature extraction a feature either exists at a given position or it doesn't.

Local features are points, edge elements, short edges or lines, and small regions. Global features comprise polygons and more complex descriptions of the image content called structures. In order to optimize the performance of FBM, features should be distinct with respect to their neighborhood, invariant with respect to geometric and radiometric influences, stable with respect to noise, and unique with respect to other

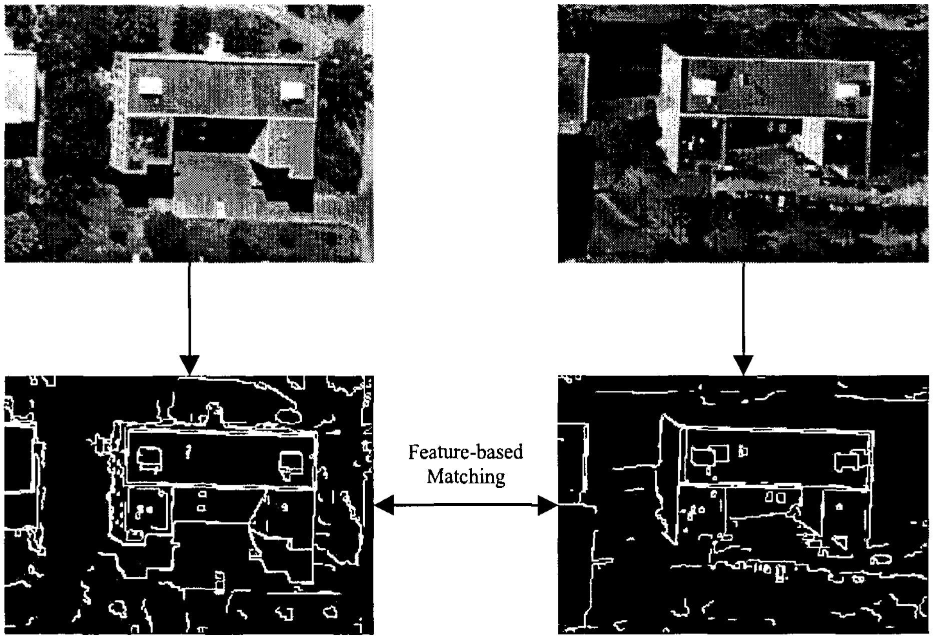


Figure 2.2: Visualization of Feature-Based Matching

features (Forstner, 1986). A visualization of the FBM is shown in figure 2.2.

Features are more abstract descriptions of the image content. As compared to gray value windows, features are in general more invariant with respect to geometric and radiometric influences. Furthermore, unlike area-based matching, good initial locations are not required before matching can begin. With geometric and/or statistical tests, the lists of features are sequentially compared and the best matches recorded (Agouris, 1992; Förstner, 1986; Grimson, 1985). Multiple and/or erroneous matches are eliminated through a global consistency check using the image coordinates of matched features and a priori information on the exposure geometry of the two images.

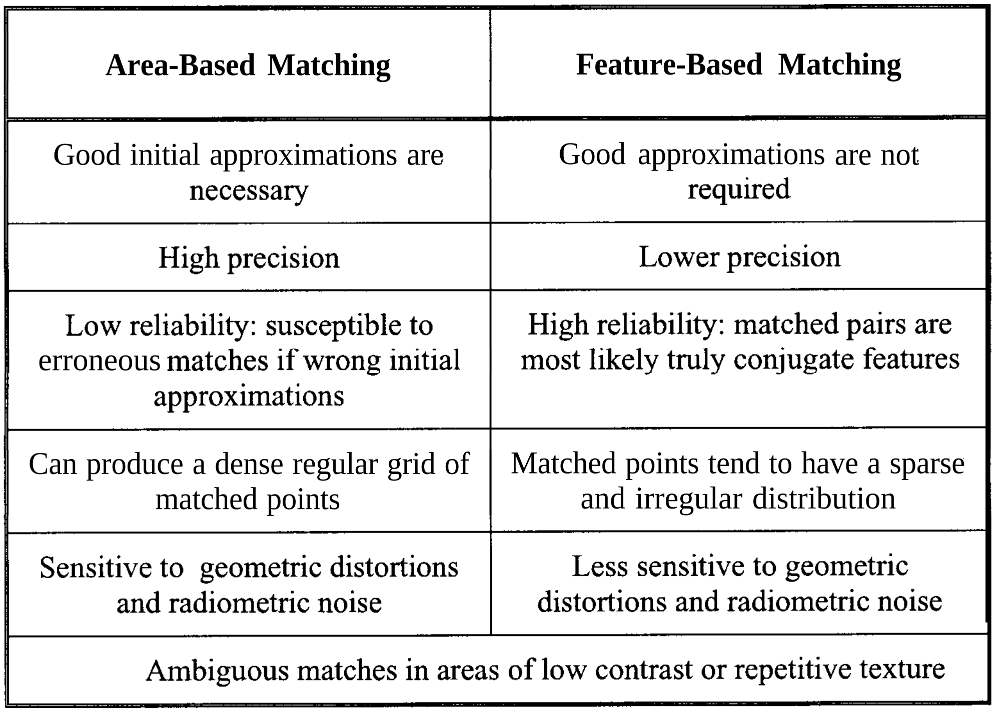
Each feature is characterized by a set of attributes. The most common of these attributes is feature position, expressed by its image coordinates. Other attributes may include edge orientation and strength (gradient across the edge) for edge elements, length and curvature of edges and lines, and the size and average brightness for regions.

As mentioned, individual features may be combined to compose larger entities, namely global features. Besides the attributes of the local features, relations between these local features are introduced to characterize global features. These relations can be geometric such as the angle between two adjacent polygon sides or the minimum distance between two edges, radiometric such as the difference in gray value or gray value variance between two adjacent regions or topologic, such as the notion that one feature is contained in another. Matching composite features is often performed using *relational matching* strategies (Shapiro and Haralick, 1987).

Feature extraction schemes are often computationally expensive and require a number of free parameters and thresholds that must be chosen a priori. Moreover, it is difficult to quantify this accuracy in metric units. In areas of low texture the density of extracted features is usually sparse. For local features, uniqueness is difficult to be achieved, and a large data volume must be handled. Global features are more seldom and thus provide a better basis for a reliable matching. However, it is difficult to define and extract global features, and they tend to be more application dependent then local features. Local features have been used for matching (e.g. Barnard and Thompson, 1980; Förstner, 1986; Hannah, 1989). In each case points were selected as features. In (Vosselman, 1992; Cho, 1995) approaches for relational matching involving global features were presented. In (Schenk et al., 1991) a combination of global and local features was used.

2.2.2.3 Comparison and Combination of Area/Feature-Based Matching

By combining the two matching methods described above, an area/feature-based matching approach can be introduced. The underlying concept is that feature detection is used, on previously oriented imagery, to identify conjugate areas rich in information, in essence providing the essential approximations for further processing with area-based matching (Carswell, 1988; Greenfeld and Schenk, 1989). This combination of the two methods is still on a hybrid level. The following table presents a direct comparison of these two techniques (Agouris, 1992).



Area-Based Matching	**Feature-Based Matching**
Good initial approximations are necessary	Good approximations are not required
High precision	Lower precision
Low reliability: susceptible to erroneous matches if wrong initial approximations	High reliability: matched pairs are most likely truly conjugate features
Can produce a dense regular grid of matched points	Matched points tend to have a sparse and irregular distribution
Sensitive to geometric distortions and radiometric noise	Less sensitive to geometric distortions and radiometric noise
Ambiguous matches in areas of low contrast or repetitive texture	

Table 2.2 : Comparison Between Area-Based and Feature-Based Matching Methods.

In our case we formulate our combined method to incorporate prior knowledge. Feature detection is not used in the sense of establishing good initial approximations, since we assume that they already exist from previous time instances. As described in Chapter 5, feature detection is used as an interpolating function while transferring the pattern over the image, to compensate for noise and include prior knowledge for the edge representation. The area-based matching step is expressed as a least squares matching problem, developed to run in a differential mode.

2.3 Summary

This chapter provided an overview of recent work in spatiotemporal modeling and image-based change detection. This overview began by describing commonly used database models. We emphasize existing work on modeling and querying in the spatial, temporal and spatiotemporal domains. Each domain's covered modeling, queries and granularity issues. One of the findings of this overview was that even though the spatial and the temporal domains have some broadly accepted theories, in the spatiotemporal area there is no general and well-accepted approach to facilitate the current needs of a multi-dimensional, multi-temporal, multi-source type geographical environment that could handle change in an efficient way.

The second part of this chapter presented a brief overview of work on change detection in the image domain. The general framework of the chosen approach (image matching) was presented, and an emphasis was placed on the analysis of the specific primitives used. Area- and Feature-based matching were presented with an analysis of the advantages and constraints of each method.

The following chapters are associated with the contributions of this thesis in the spatiotemporal field. We begin with chapter 3 that provides an overview of a new spatiotemporal model.

Chapter 3

Modeling Change

Understanding how objects evolve over time is a fundamental issue in modeling the real world in an appropriate manner. Change and the processes of change need to be formalized in order to create a useful spatiotemporal model. In current geographic information systems only a small fraction of potential information resources is handled. A relatively small amount of information is synthesized into map form. It is common practice to have multiple sources of information containing multiple representations of features that are reduced to a single representation. This process can eliminate useful information and particularly information on change. Accordingly, these systems are unsuitable for today's applications, where geospatial information becomes increasingly dynamic and spatiotemporal in nature.

In developing a model of change it is important to first consider the evidence or basis for identifying change. In this thesis change is extracted through comparisons of observations. Observations are mostly concentrated in the image domain and can include satellite images, aerial or close-range photographs and video sequences. In addition to that, other sources are incorporated to facilitate change detection, such as vector and attribute data. Collectively these heterogeneous observations form the evidence for identifying change through a novel image matching technique (Chapter 4).

In this chapter we present a framework for the integration of digital images and complementary GIS datasets in a model that provides explicit information about spatiotemporal change. We make use of the SpatioTemporal Gazetteer (Beard and Agouris, 1997) as a model that handles multiple information. Modifications on this model are introduced based on specific needs that result from the image-based change detection. Moreover, an expansion of the model to a broader spatiotemporal environment is presented. The general components of the expanded model are shown and interaction with the change detection analysis is discussed.

3.1 SpatioTemporal Gazetteer Model

In Beard and Agouris (1997) the *SpatioTemporal Gazetteer* (STG) was proposed as a new model that makes more effective use of multiple information resources. This spatiotemporal model is roughly based on a digital library model (Beard and Smith, 1998; Beard et al., 1997; Goodchild 1998). The approach shares some of the characteristics of the multimedia geographic information system, proposed by Lombardo and Kemp (1997) and symbolic description of image sequences using spatiotemporal logic (Del Bimbo et al., 1995). The model consists of a repository of heterogeneous information sources, with a set of indexing structures to organize and access them (Fig. 3.1).

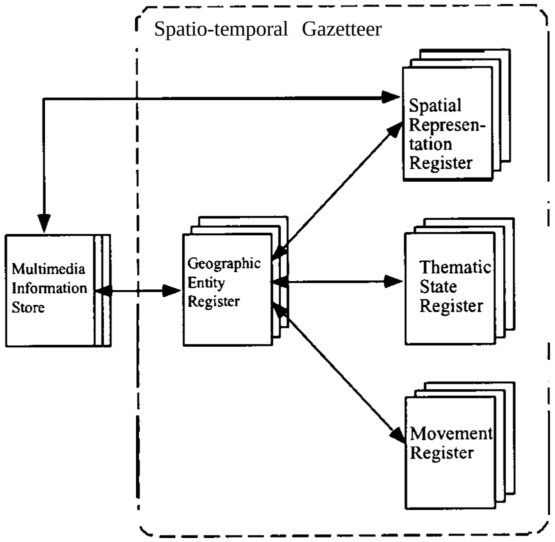

Spatio-temporal Gazetteer
Spatial Represen-tation Register
Multimedia Information Store
Geographic Entity Register
Thematic State Register
Movement Register

Figure 3.1: Conceptual Model of Integrated Spatiotemporal Gazetteer and Multimedia Information Store (Beard and Agouris, 1997).

3.1.1 Multimedia Information Store

The repository of information sources is referred to as the *multimedia information* store (MIS). It can include imagery, maps, video, text documents such as books, newspaper reports, and magazines, and scientific data sets such as meteorological or oceanographic observations. The goal of the MIS is to provide a continually growing repository with new information potentially added on a daily or even more frequent basis. This repository need not be a single site, but could be a multinode distributed repository (e.g., NSDI clearinghouse nodes).

The multimedia information store is characterized as an *implicit* information store since there is a large volume of information in an individual information source such as a satellite image and even greater volumes in combinations of information sources (e.g., maps and images). Individual units within the information store are referred to as information objects.

3.1.2 Spatiotemporal Gazetteer

The *SpatioTemporal Gazetteer* is an indexing structure over the multimedia information store and is the key mechanism for representing latent information contained in the multimedia information store as change information. Components of the gazetteer store explicit change information. The gazetteer consists of several subcomponents that include:

- a geographic entity register,
- a spatial representation register,
- a thematic state register, and
- a movement register.

The *geographic entity register* maintains a record of identified spatiotemporal entities. The *spatial representation register* like the *thematic* and the *movement* registers, is a collection of relations between geographic objects and individual information objects from the information store. This register tracks the multiple spatial configurations that are possible for geographic objects. The *thematic state register* tracks states of geographic entities as provided by specific information objects. States are multidimensional so a geographic object may have several concurrent states. The *movement register* is a collection of relations between geographic entities and pairs of information objects and it tracks movement. A more detailed description of each component can be found in (Agouris et al, 2000c).

3.2 Spatiotemporal Gazetteer Modifications

The major goal of this thesis is to detect changes in digital imagery. In order to integrate this image analysis technique with a spatiotemporal environment, a model should be introduced. We make use of the SpatioTemporal Gazetteer (Beard and Agouris, 1997) as described in Section 3.1 and we introduce modifications on the existing model based on specific needs that result from the image-based change detection.

3.2.1 Space over Time or Time over Space?

A spatiotemporal object is dynamic and the appropriate reference system should be created to project spatiotemporal information. Strictly speaking, in the spatiotemporal domain we cannot have spatial objects that do not have a temporal representation, and temporal objects that do not have a spatial extent. In spatiotemporal GIS applications the spatial parameters are observed over time. So the temporal domain is acting as the reference system for the spatial one. In such systems and in order to retrieve spatial information we should establish mapping functions that project the temporal domain over the spatial one, so for one temporal instance we can have the corresponding spatial instance.

Following the majority of GIS approaches, within our spatiotemporal environment change is addressed as the evolution of space over time and not time over space. In order to justify this, let's assume that we use as a reference the spatial domain. We would need to create functions (F_t) that project time (T) over space (S), so in one spatial instance, we know the corresponding temporal instance. In addition to that, spatial constraints (s_1, s_2) for that function should exist showing its spatial extent. The mapping function is formulated as follows:

$$T(S) = F,(S), \text{ where } s_1 < S < s_2 \qquad (3.1)$$

The problem in this case is that this function is one-to-many because one input can result to many outputs. One spatial property can be the same over a time interval, so going from the spatial domain to the temporal would not give us a unique solution. For example the position of a car might be the same in two temporal instances, but if a spatial reference system is used, we cannot know to which of the two times that spatial instance corresponds. Therefore even if we know the input, the output cannot be calculated in all cases.

The other approach would be to use time as the reference system. Here we should note that time is considered as a linear, constantly increasing function. Our system treats time as it is in reality, linear and continuous. Other time models, such as the cyclic one, are conceptual models of time, efficient in some cases to implement, but not appropriate for a general spatiotemporal framework. If the issue of a cyclic model arises, then that model should be backprojected in reality and then inserted in our proposed model.

So the relation (F,) that projects space (S) in time (T) for a time interval (t_1,t_2) would be:

$$S(T) = F,(T), \text{ where } t_1 < T < t_2 \qquad (3.2)$$

By examining the above function, we can realize that the problem mentioned above is solved. For one time instance we know the unique corresponding space instance. For the example given above only one position of the car can correspond to a specific time instance. Due to this *many-to-one* relationship, spatiotemporal relations are addressed as spatial relations over time.

The S(T) function expresses the relation between time and space. Our goal though is to describe changes of space over time. So we need to define new functions that express that. One way to do this would be to differentiate the above function so we come up with:

$$AS(T) = \frac{\partial S(T)}{\partial T} = \frac{\partial F_s(T)}{\partial T}, \text{ where } t_1 < T < t_2 \qquad (3.3)$$

where ? S(T) is the spatial change function over time. Another approach would be to define new functions that describe directly the spatial change over time ($F_?$ s):

$$AS(T) = F_{,,}(T) \text{ , where } t_1 < T < t_2 \qquad (3.4)$$

In either case, within our model, time is the chosen reference system for spatiotemporal change.

3.2.2 Change-Oriented Design

As it was discussed earlier for each spatiotemporal object the temporal domain is used as a reference system. The question is how a spatiotemporal object can be represented within a spatiotemporal model. Currently two database design approaches are available for handling spatial data - the relational and the object-oriented (Fig. 3.2). The debate in the research community is still very active on which model is more appropriate. For our case which handles spatiotemporal data the object-oriented model is chosen. With this we try to model change that appears in the spatiotemporal space through an aggregation of changes that appear in discrete spatiotemporal objects over that space. This approach provides some advantages over the relational model, because:

- The object model is consistent with our change detection approach (Chapter 4), where spatiotemporal changes are expressed through changes on discrete objects.

- Objects can be considered as instances of classes, which allows behavior modeling.

- Class inheritance and propagation extends querying capabilities with multi-resolutional navigation.

- The object model allows complex objects to be attribute domains; this is prohibited in the relational model.

- The object model is much easier to implement from the software engineers point of view and provides high capabilities for customization.

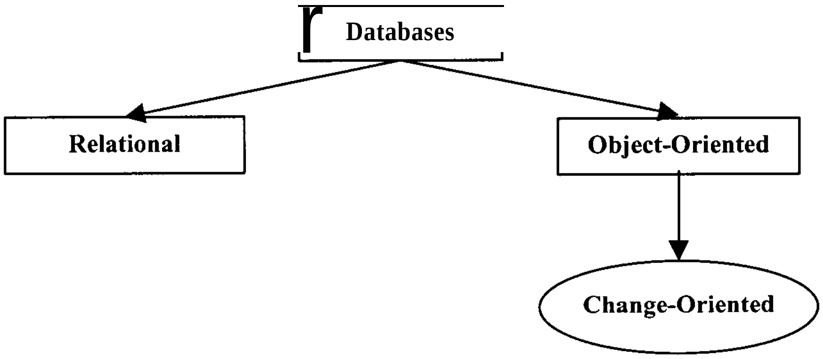


Figure 3.2: Database Models

One of the major goals of each database design is to ensure minimal redundancy and maximum efficiency. In a traditional approach using object-oriented design representations of objects are stored from different time instances. In this case change would be handled in an *implicit* way, in essence calculated and not stored. This would have a significant impact in the two optimization factors mentioned above. First, a significant amount of redundant information would be available, since the same representations of the same object would be stored. Second, the critical communication system would have to compute change each time, decreasing considerably performance.

To compensate for the problems arising from object representations, this design model is extended to perform in a differential mode creating a change-oriented approach. This model is still an object-oriented one, but primitives (objects) for this approach are *the changes that appear on observed geographic objects.* Reasoning for the chosen approach is found in the efficiency and redundancy areas, respectively, as described in the following two sections.

3.2.2.1 Maximum Efficiency

Let O_j be an object from the set of objects (0), and that object is observed in a subset T_n of the set of time instances (T). So we have:

$$O_j \in O, T_n \in T \text{with } T_n = [0, t_1, t_2, t_3 ,..., t,] \tag{3.5}$$

We claim that the representation of an object at a time instance t_n can be expressed as the accumulation of changes that appeared from the time that the object was created until time t_n.

$$O_j^{t_n} = \int_0^{t_n} \partial O_j \tag{3.6}$$

If we disintegrate the above integral based on the subset T_n, we have:

$$O_j^{t_n} = \int_0^{t_1} \partial O_j + \int_{t_1}^{t_2} \partial O_j + ... + \int_{t_{n-1}}^{t_n} \partial O_j \tag{3.7}$$

With this we represent an object in time through the accumulated changes that appeared in that time. We store change directly in an *explicit* way and not *implicitly* by saving different representations of geographic object through time. A very important intermediate step can be introduced between object representations and change detection. Through reasoning, object changes can be distinguished from different representations of the same, non-changed object. This contributes to minimizing redundancy, as we will see later, but mostly provides validity for the stored representations.

Maximum efficiency for handling spatiotemporal change is achieved with this change-oriented approach, since change is directly available within our environment. Change information is retrieved quickly, without any computations, permitting real-time communication times. If change is not what the user wants, but a state of an object at a specific time, even though this is not the objective of this proposed environment, we still allow that operation implicitly by reconstructing object states through change accumulations.

3.2.2.2 Minimal Redundancy

Let's assume that change exists in a multi-dimensional set R^{P}, where p is the number of dimensions. If we apply that on the equation (3.7) as far as dimensionality is concerned we have:

$$ {}^{p}\left|O_j^{t_n}\right. = {}^{p_1}\left|\int_0^{t_1} \partial O_j\right. + {}^{p_2}\left|\int_{t_1}^{t_2} \partial O_j\right. + \ldots + {}^{p_n}\left|\int_{t_{n-1}}^{t_n} \partial O_j\right. \qquad (3.8) $$

$$ \text{where } p_1 + p_2 + \ldots + p_n \leq n * p\,. \qquad (3.9) $$

This equation is the key to providing minimal redundancy in our system. We save only change information, and more specifically only the dimensions of change that are modified. With this approach we can always ensure minimal redundancy by reducing a multi-dimensional problem to its minimal modified dimensions.

Because no change is still considered change information, a detailed description is stored for the dimensions that changed, and an indexed change information structure is created for the dimensions that were not altered.

3.2.2.3 Decomposing Change

In order to model change in a complete way, a framework should be established to include all its aspects. The previous section presented the need for handling a multi-dimensional problem based on its minimal modified dimensions to ensure minimal redundancy. In order to achieve this, a multi-dimensional change decomposition is performed. The ultimate goal is to express the multi-dimensional change by a set of 1-D elements, in essence the attributes stored in our database(s).

Let’s assume that change exists in a multi-dimensional set R”. We decompose change as an aggregation of j subset spaces,

$$R^n = [C_1^{a_1}, C_2^{a_2}, \ldots, C_j^{a_k}] \qquad (3.10)$$

where $C_1, C_2, \ldots, C_j$ define multidimensional subspaces of change, and $a_1, a_2, \ldots, a_k$ are the corresponding dimensions of each subspace. In a way these subspaces act as the bases of the multi-dimensional change space (R^n). It should be noted that optimally no overlap should exist between these subspaces, so

$$a_1 + a_2 + \ldots + a_k = n \qquad (3.11)$$

meaning that the same change dimension cannot be expressed by different subspaces.

By applying the same decomposition for each non-one-dimensional subspace, for example $C_1^{a_1}$ with $a_1 \neq 1$, we have:

$$C_1^{a_1} = [C_{11}^{b_1}, C_{12}^{b_2}, \ldots, C_{1i}^{b_m}] \qquad (3.12)$$

where C_{1i} are the lower dimensional subspaces that each subspace is further decomposed into. Similarly to the previous decomposition, b_m are the dimensions of those subspaces, with

$$b_1+b_2+. . . +b_m = a_1 \tag{3.13}$$

to ensure validity of our decomposition. We keep on decomposing each subspace until we come up with 1-D subspaces, which are the attributes of change.

As previously mentioned, optimality is achieved if minimum overlap exists. That does not mean that there should be no correlation between the 1-D subspaces. For example the material of the road might be related with the width of the road. And the same can be claimed for the shape of the car (e.g. open doors) and its speed. Another issue that should be discussed is the relation of the change attributes stored in our system and the actual changes that appear in reality. For the example of a car, what is observed in the real world when a car moves is the change at the X,Y,Z coordinates of the parts (molecules) that form the car. In most databases, it is meaningless to store the movement of the car in such a way. Thus, conceptual representations of this movement are created, for example the movement of the center of gravity. Moreover, the sets of coordinates for the center of gravity can be replaced by types of functions describing the path. Within our model we generalize reality by this set of 1-D elements of change. They can be detailed, as in the center of gravity, or more abstract, as saving the type of path (e.g. linear, curve). Therefore, with this approach it is up to the user and the application to define their own dimensional configuration that will act as the bases of change.

3.2.3 Conceptual Implementation

The theoretical issues that should be taken under consideration were presented above. In order to address them within the context of our model we extend the existing Gazetteer model by introducing a *Differential Spatiotemporal Gazetteer* (DSTG). The term “differential” is used to emphasize the purpose of this design, which is to handle change explicitly and not implicitly through different representations of objects. We extend the model to handle change primitives directly following a *change-oriented* approach. We store an initial state of the object (in essence change from non-existence) and subsequent changes. An object representation at time t^{n} is obtained as the aggregation of t^{0} and all subsequent changes ? $t^{i-1,i}$, for $i=1,. . . n$.

Our approach is derived from the need to manipulate objects that *“can be represented and identified in the image domain”.* The common factor in this information is that some type of space-time coordinates exist. Either one or both of these coordinates may be precise or, to a certain degree, tizzy. For example, the duration of a river outline change (e.g., flood) may be difficult to pinpoint, and the same may be said about its spatial extent. The objects that are examined are restricted to physical objects such as roads, buildings, cars, islands, lakes, rivers and land parcels. Conceptual objects (entities) are beyond the scope of this thesis (e.g. countries). Therefore the use of geographical objects refers to physical objects.

3.2.3.1 Geographic Identity Information Level

The Geographic Identity Information Level (GIIL) in the DSTG corresponds to the geographic entity register as presented in the original STG. The content of this conceptual level of information is modified to express physical objects and not entities. The attributes remain the same, but address objects that can be identified in imagery.

3.2.3.2 Child Information Level

The gazetteer comprises sub-components that describe property changes on specific objects. The objective is to have minimal overlap between components, while maximizing their aggregate content. In the pre-existing STG model, three multi-dimensional groupings are recognized, forming the spatial representation register, the thematic state register, and the movement register. Extending this classification we claim that it should not be fixed but rather flexible. Depending on the application, change is perceived in different ways. Some attributes might be of interest, some others not. Even the same attributes might be handled in a different way. For instance, the color of an aircraft in an infrared image might be classified as a thematic attribute, or a movement attribute since it betrays recent or future movement. In a higher reasoning level, different groupings of attributes may be introduced to facilitate specific needs. A video sequence of a man rotating his hands could be interpreted as:

- shape deformation (e.g., by an athletic trainer watching the muscles of an athlete),
- movement (e.g., by a driver watching a car traffic controller), and
- thematic change (e.g., by a doctor watching his patient’s face color).

To compensate for the multiple interpretations of change, we propose a more general framework that expands the three predefmed registers. We express actual characteristics of change at the object level through a collection of *child information* *levels.* Change is stored directly and representations of objects are reconstructed through change accumulation. Each child collection corresponds to a specific dimension subset of change with the attributes defining the one-dimensional sets of the final change decomposition. It can inherit the form of a traditional object-oriented database, with the difference being that the attributes express change in a discrete mode. In a more advanced case, functions can be stored that express change in the attribute(s) in a continuous way where that is feasible.

This structure provides to the user the flexibility to assign properties of interest and in a more abstract level conceptually group these properties. In doing so change information is filtered in two ways. First, redundant information is disregarded, and second, change is expressed in our environment through the user eyes. As shown in the above example of a moving man, parameterization of change is essential in order to compensate for the different perceptions of reality and in essence change.

The selection of the child information sources can be based on different criteria, depending on the application. Optimality is achieved in defining child information source if:

- independence between them exists to ensure minimal redundancy, and
- completeness to describe all essential properties.

Accordingly, an optimal segmentation of the child information space will approximate semantically a base in the object’s change space. However, optimality is not a necessary condition when deciding the content of these child sources. Often, pre-existing constraints (e.g. standards and practice conventions) dictate the assumption of specific child structure.

As mentioned before, the child information level expresses grouping of dimensions in a conceptually meaningful way. If each child source represents n change dimensions, then the final dimensionality of that source would be n+1, since these dimensions have to be projected on the time reference system. The final choice of grouping depends on the application and the physical objects under examination.

Applying the change-oriented approach to the existing STG three child registers would be created corresponding to the spatial representation register, the movement register, and the thematic state register. Within these child registers the following modifications should be made to compensate for the fact that we are storing change information explicitly and not implicitly through different object representations:

A) The spatial representation register would now store changes in the spatial representation of the object as extracted from the information object.

B) In the movement register changes in shift and/or rotation information indicating movement in the 3-D space would be stored.

C) Similarly to the previous two, the thematic register would record changes in, instead of multiple versions of, objects.

3.2.3.3 Change Indexing Information Level

As mentioned before, in the DSTG, change is represented explicitly in the model. Another major advantage of this approach is the ability to create indexing structures that directly express change. To benefit from this, a new component is introduced in our DSTG, the *Change Indexing Information Level* (CIIL). This level provides the essential multi-dimensional indexing mechanism (Langran, 1993), with flags to all the attributes of the child sources that have been modified. In other words this indexing structure is showing whether change has occurred in a specific dimension. This allows fast retrieval of change information, especially when the user is not interested in a detailed description of change. More information of this information level is provided in Section 3.3.2.

3.2.3.4 Differential Spatiotemporal Gazetteer Structure

The structure of the DSTG is shown in figure 3.3. Our DSTG is composed of a *Geographic Identity Information Level*, a *Change Indexing Information Level* and a number of *child information sources.* From the change dimensionality point, the DSTG can be considered the space of extracted change, the child sources being the change subspaces, and the instances in these databases the 1-D subspaces expressing change. The indexing mechanism in the 1-D subspaces (attributes) is the *Change Indexing Information Level*, showing whether these change elements are modified or not. A more general description of change, addressed at the object existence level, is expressed through the *Geographic Identity Information Level.*

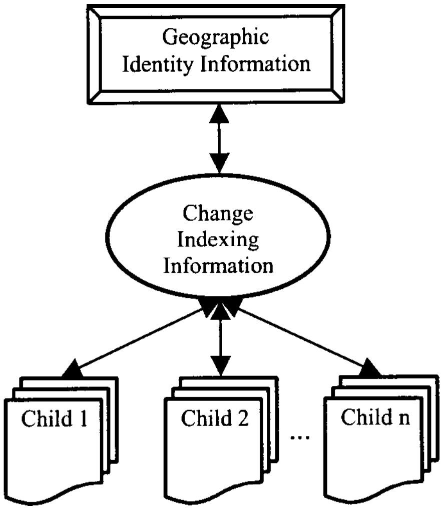


Figure 3.3: Differential Spatiotemporal Gazetteer Structure

3.2.3.5 Mathematical Dependency between DSTG Components

The next step in our modeling is to explore relations between the components of the DSTG. These relations can be explored if we start from the child information level and move towards the more generalized indexes. Let's assume that one of the properties we are interested in is the size of a house. We observe this attribute, and after our change detection the graph of figure 3.4 is created, showing the function of size over a time interval. Since DSTG saves changes in objects, the corresponding values stored in the child level are represented through the first derivative of the size function (Fig. 3.5).

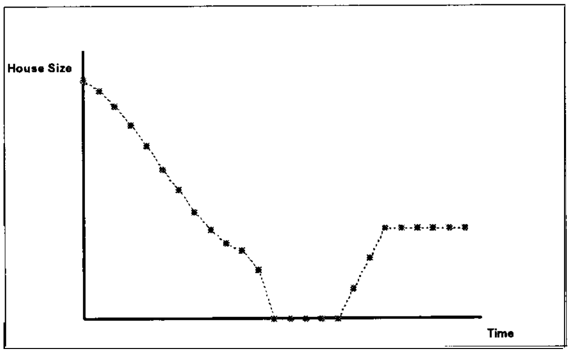


Figure 3.4: Expressing Objects Property (House Size) through Time

The nodes with larger marks denote that there is no change, so actually nothing is

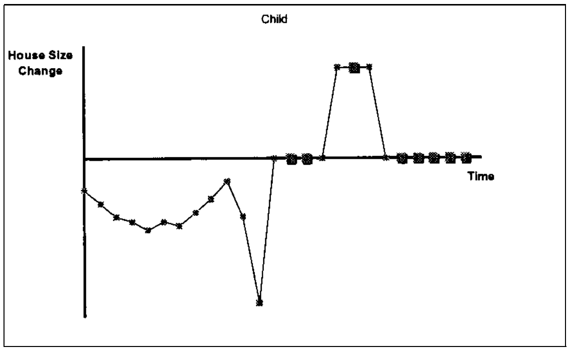


Figure 3.5: Expressing Change at the Objects property (Change in House Size) through Time

saved at the child level, except a pointer to the CIIL. As we mentioned, CIIL shows if an attribute of change has been modified or not. If that attribute is a quantitative one, like the size of a house, then this corresponds to a boolean operator on the first derivative of the size function. If the first derivative is zero (on observed, not interpolated values) then that means the "change attribute" is zero, so no change exists. Otherwise, a modification has taken place (Fig. 3.6).

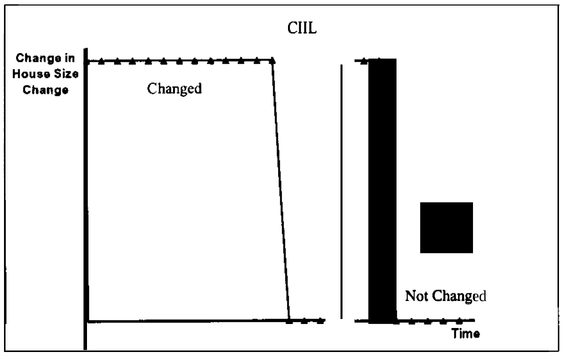


Figure 3.6: Expressing Existence of Change at the Object's Property (Existence of Change in House Size) through Time

We can also introduce reasoning in this process and correlate the existence of an object with its attribute values. So for the above example, the GIIL representation for that object over that time interval is formulated accordingly (Fig. 3.7). Where the size of the house is zero then that object stops existing, otherwise continues to exist. Interpolation issues between the observed points are not addressed in this analysis.

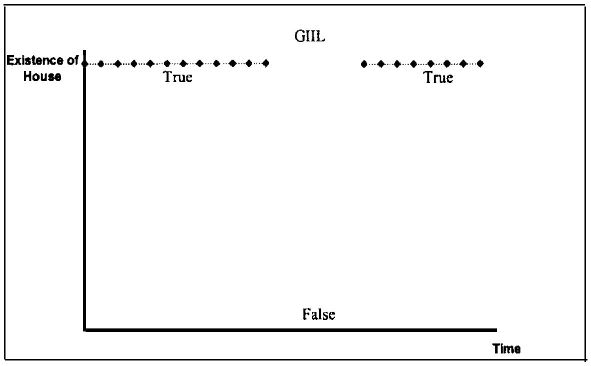


Figure 3.7: Existence of House as expressed through the House Size over Time

3.3 Granularity Issues in Querying and Designing the Differential Spatiotemporal Gazetteer

A significant functionality of a spatiotemporal model is the ability to express explicitly changes that may be found implicitly in various sources and object representations. Defining a common measurement framework to express spatiotemporal change is a challenging task, as meaningful changes vary according to level of detail, nature of application, and type of object. Furthermore, the context of change, as determined by applications and user needs and queries, can be considered to be spatial in nature, temporal, or a combination of both. The communication of this change requires the consideration of users who have diverse needs, and exhibit various levels of understanding of the content and organizational schemata of the databases that they query.

In (Mountrakis et al, 2000) the problem of spatiotemporal database queries is addressed by *assisting user navigation in multiple spatiotemporal granularities.* In this paper a novel concept in detecting and propagating change, the *hierarchical classification of change resolution* is introduced. A dynamic classification scheme is presented to support change monitoring and communication of spatiotemporal objects. A sliding rule paradigm is employed to support navigation through multiresolutional data. With this approach the concept of granularity is extended in the query process of a spatiotemporal model.

In the following sections an intuitive generalization of change in levels of detail is presented and its interaction with the DSTG is explained. We introduce a query translation that specifies the input datasets for the change detection. Further analysis of this spatiotemporal querying method is beyond the scope of this thesis.

3.3.1 Categorizing Queries by Level of Detail

In a querying process, the user is trying to extract specific information from a model representing the reality. However, in every modeling attempt a generalization of reality is performed. One of the keys to successful modeling is how close it is to the user needs. Projecting this concept to the application of queries its translation would be to return information as close to the desired information as possible. In a complex environment though if a fixed generalization is chosen the environment is significantly constrained. That does not mean that a filtering procedure for disregarding too specific or too general information should not exist. But within this ‘kesolutional” framework the user should have the ability to define his/her own resolution and extract information meeting his/her interest.

In order to achieve that a general classification for change is proposed. This classification as described in (Mountrakis et al, 2000) acts as the tool that will allow the navigation of users in multiple resolutions. In this intuitive classification, we introduce a hierarchical structure to categorize spatiotemporal queries according to their level of detail. The backbone of this hierarchy is based on the following three levels of change.

1) Existence of Object

At the first level, a boolean query can be performed to detect whether an object exists or not (Query Level 1). This is a single-source query, which can be applied repeatedly, but there is no dependency between the sources used. At this level, the object is treated as a black box with no further spatiotemporal attributes. This type of query has been referred to as identity-based query (Hornsby and Egenhofer, 2000).

2) Existence of Object Change

At this level, the user can have a simple Yes/No answer to a query whether any evidence of change (other than existence) for a specific object has appeared. This query is differential in nature, since two or more sources are required. Queries of this type (Hazelton, 1991) examine objects in a more detailed way, but still particular change attributes are not considered.

3) Characteristics of Object Change

This is the most detailed level, where a complete description of all the attributes of the object that changed is provided. As this is also a differential query, at least two sources are needed.

3.3.2 Dependency of Database Design and Levels of Detail

As described in Section 3.2, the proposed gazetteer model consists of a *Geographic Identity Information Level* (GIIL), a *Change Indexing Information Level* (CIIL) and a number of *child sources* at the child information level. One of the reasons that this structure was chosen is based on the need to support spatiotemporal queries at different levels of detail.

At the coarser level, the database stores information about existence through time in the *Geographic Identity Information Level.* Queries of type 1 (see previous section) can be addressed at this level (GIIL, Query Level 1 of Fig. 3.8). Queries of type 2, which

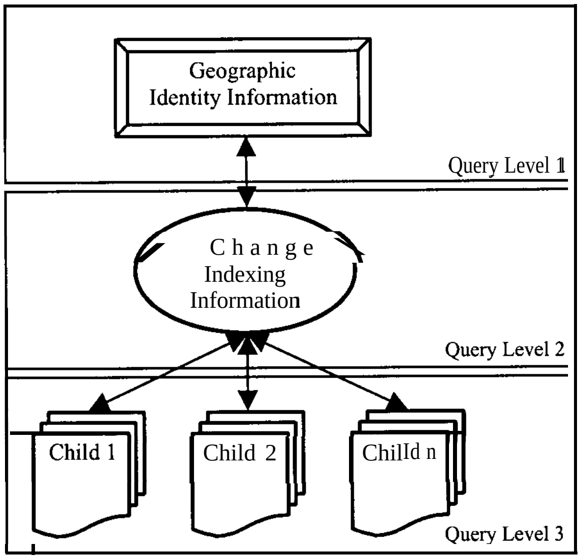


Figure 3.8: Dependency of Database Design and Levels of Detail

involve the existence of object change, can be resolved only by accessing the Change *Indexing Information Level* (CIIL, Query Level 2 of Fig. 3.8). If a detailed description of object change is requested then the *child sources* are accessed (child, Query Level 3 of Fig. 3.8).

It should be noted here that the above model structure and query configuration is assuming a multi-resolutional approach. If the goal of a specific spatiotemporal environment is to model specific changes at specific resolutions from specific datasets and the querying resolution is fixed, then the above structure would significantly increase the cost of application and decrease performance. On the other hand, if a multi-resolutional method is chosen, the benefits of this model structure are evident, as explained below.

The goal behind the chosen architecture is to limit access to the child sources, and improve query response time. This is critical in the case of a multi-node approach (Intranet, Internet, etc) or even larger organized distributed environments, like the National Spatial Database Infrastructure. It enhances performance, since in a number of cases only indices should be accessed, without the need to interact with the more time-consuming child sources. So for example if the query requests change information on a car's X coordinate, the information flow will follow a coarser to finer resolution path. First the GIIL will show whether this car still exists, and if it does the CIIL will be accessed. Based on the indexing structure for this property, information from the appropriate child source(s) will be retrieved.

The correspondence of the three query levels to specific information levels is the general approach followed in this thesis. Nevertheless, the query levels do not necessarily have to point to specific information level. A more detailed analysis can be performed to relate attributes of the child level, or group child sources together. More complex structures can combine attributes from the CIIL and the child sources. This exploration is left for future work.

3.3.3 Spatiotemporal Granularity Metrics

3.3.3.1 Minimum Spatial Element

In order to establish a resolution reference for change in the spatial domain the term of the Minimum Spatial Element (MSE) is presented. The MSE expresses the spatial resolution of the recorded change and can have an absolute or a relative value. For example, the MSE of the building class can be a cube of lm x lm x 1m volume. In this case, only changes larger than this element are returned to a query on whether something has changed in a building object. But the real flexibility that MSE provides is when a relative value from a tree structure expressing granularity of change in the spatial domain is assigned. For the same example, the building class can have as a MSE the wing, room, or brick class. If one expands this to other object types (classes) that might exist in the database (e.g. town class) and consider the building class as a MSE for the town class, then automatically a propagation process for change resolution is triggered. A more thorough description of this approach can be found in (Mountrakis et al, 2000).

3.3.3.2 Minimum Temporal Element

Extending the MSE definition to the temporal domain, a predefined time interval that expresses the minimum duration of a change is introduced. This temporal interval is the Minimum Temporal Element (MTE). It is user-defined and describes the maximum temporal resolution of the users interest. Therefore we can reasonably claim that the continuous model of reality can be projected successfully on a snapshot model without any loss of information (for that specific user).

3.3.4 Spatiotemporal Query Translation

The levels of detail (section 3.3.1) and their implicit classification interact in a query process. They do so by decomposing the query into three parameters, namely

- level of detail,
- spatial extent, and
- temporal extent

and by transforming the query into a range of resolutions.

Query [Level of Detail, Spatial Extent, Temporal Extent]

The level of detail can have values of 1, 2 or 3, based on the three levels described before, namely Existence of Object, Existence of Object Change, Characteristics of Object Change. The intermediate step of translating a query in the above format is beyond the scope of this work.

The spatial and the temporal extent of a query help defming the spatiotemporal area of interest. These extents also help the user to define the MSE and MTE values and propagate change in the spatiotemporal domain (Mountrakis et al, 2000). The use of this query translation is described in Chapter 4, where the proper datasets are chosen for the image-based change detection technique.

3.4 Expanding DSTG to a Broader Spatiotemporal Environment

In the originally proposed model there is a relation between the MIS and the STG, showing the connection between implicit and explicit change information. In order to perform this transformation the essential processes should be introduced. This was the original goal of this thesis namely to develop processes that detect change in imagery. However, these processes cannot be isolated, but should rather interact with the source (MIS) and target (DSTG) change information. Therefore, a new component is introduced, the *Change Detection Tools.*

Furthermore, STG stores information on observed changes. Additional information should be included on modeling behavior aspects of change. This information would allow validity checks on the STG content, prediction based on behavior patterns, and would facilitate future change detection attempts. Based on the above, we expand the model by introducing another component, the *Knowledge Base.*

By inserting these two components and showing their interaction with the MIS and the modified STG, the DSTG, a *broader environment* that handles change is introduced (Fig. 3.9). From now on we treat the proposed DSTG not as the spatiotemporal model itself, but as one of its components. Further details and justifications for the establishment of this broad environment are presented in the following sections.

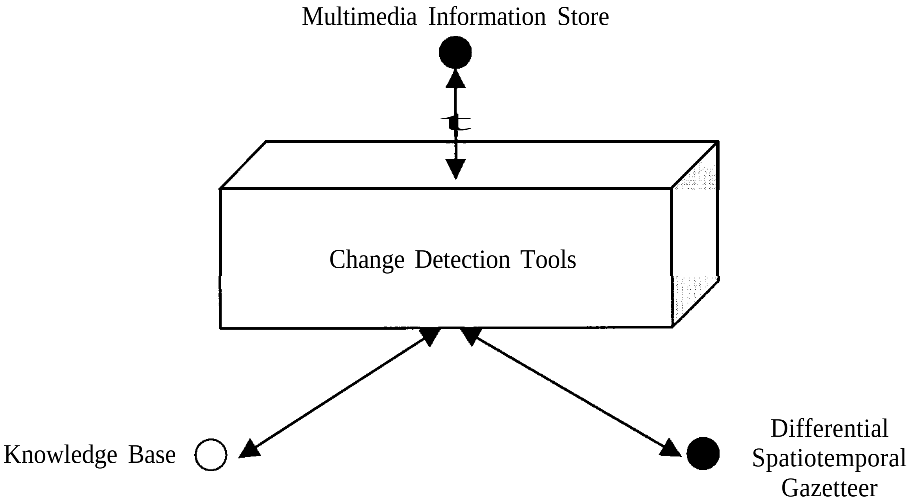


Figure 3.9: Extending SpatioTemporal
Gazetteer to a Broader Spatiotemporal Environment

3.4.1 Types of Change

In order to further analyze change, we should consider different categories for change information. For the purposes of our classification, a basic distinction is introduced in spatiotemporal change: the *potential* of an object to change, and the *actual* or *observed* changes. For our formulation we follow a Java object-oriented paradigm, where each programming object is defined as a combination of state, behavior and identity, with:

- state expressing data elements and their value,
- behavior showing what the object does as methods are called, and
- identity ensuring uniqueness through identifier and attributes.

3.4.1.1 Observed Changes

The *observed* changes provide the essential framework for actual change information to be stored and retrieved in an efficient way. Information of this type is stored in the DSTG (Fig. 3.10). It can be further analyzed into *identity-based* and *state-based* changes. For the purpose of our modeling, *identity-based* change is treated as existence or non-existence information (Hornsby, 1999), without any further processing. It is addressed by the content of the GIIL within the structure of the DSTG. Here we should note that modeling of this type of change is beyond the scope of this work.

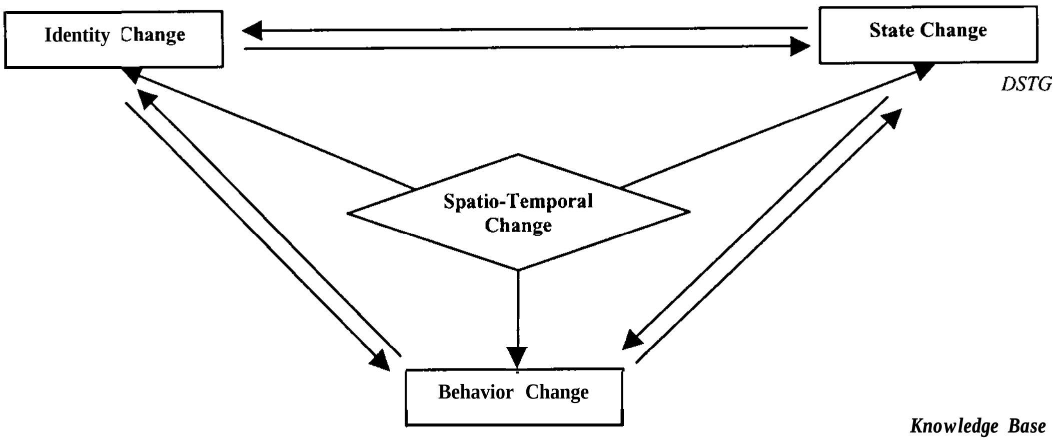


Figure 3.10: Types of Spatiotemporal Change

The *state-based* change gives a detailed description of all the aspects of change (beyond existence) that occur at the object level. These changes are expressed through the collection of child sources and the child information level. As we saw earlier sometimes there is a reasoning relation that can be established between the identity and the state changes. For instance when the size of the house is zero, then the house is assumed to no longer exist. This is a two-way relation, since if the house does not exist, then its size will be zero.

3.4.1.2 Predicted Changes

The *potential* of an object to change is expressed through *behavior-based* change information. Behavior based change describes different ways, patterns, constraints and conditions of change evolution. This type of change is a separate, distinct category but it can result from the actual changes mentioned above. Any updates either at the GIIL or the child level should trigger a two way communication. One way ensures validity in the DSTG contents based on the behavior constraints, and the other way updates behavior modeling based on new change evidence.

Behavior information is vital for the performance of the change detection techniques, and can provide validity for the obtained results stored in the DSTG. For example, if raw change information for a car's movement is stored in a child source, then interaction with behavior models could provide functions describing the objects path, and allow future predictions for its position. This would be interpreted as the searching area for the change detection algorithm, providing higher performance and accuracy in future detections. And change information could be filtered based on specific constraints that may appear on the object behavior, for example a car’s speed cannot be higher than 150 mph. The same interaction could be established between the GIIL and the behavior models (e.g. if the width of a road is smaller than 30 feet, then the object highway might not exist anymore).

3.4.2 Knowledge Base

All these constraints and interpolation models depend on the application and forms of change that might be of interest. This type of change information is not included in the DSTG, where only observed changes are stored. Yet, the necessity of including such information is evident. Based on the above findings, we expand our model by introducing another component, the Knowledge Base. Note that the focus of this thesis is not to provide a comprehensive description of the knowledge base, but to contribute to the establishment of the general framework for the development of such a component and specifically make use of its content in the change detection process.

The main objective of this component is to model spatial behaviors over time, since time is our reference system. The diversity of these behaviors has been pointed out in the past in (Langran, 1993; Couclelis, 1996). Although behavior models can be created for each dimension of change, a more global approach is used so that change behavior can be grouped into more meaningful ways. As a general example we could take the original three child registers presented in the original STG. Following that grouping, one characterization of a spatial object's behavior is its ability to move over time. Objects can range from strongly fixed in a location (mountains, lakes) to partially fixed (rivers) to partially mobile (plant species) to fully mobile (people, animals, vehicles). Another dimensional grouping of object behavior over time is an object's ability to change its geometric shape. Some objects may have relatively plastic behavior whereas others will be relatively rigid. A third dimension of behavior is an object's ability to change its non-spatial or thematic characteristics over time. Some attributes of a phenomenon will be relatively constant while others will be highly variable. For example, an animal has the characteristic of having fur which is constant over its life span while its weight may be constantly fluctuating. A link in a water distribution system can be characterized as having a constant pipe dimension but its flow will vary.

These three multi-dimensional spatiotemporal behavior groups can be expressed as mobility, elasticity and thematic variability, respectively. In a more advanced level, these groups can interact in complex ways. Examples of objects which are strongly fixed in a location and show little geometric variability with time are mountains and rock outcrops. Objects which are strongly fixed in location but vary their geometric shape substantially with time include many hydrologic phenomena such as lakes and wetlands. Some of these behaviors may be highly dependent on temporal cycles while others will be relatively independent of temporal cycles. Train routes and seasonal animal migrations are examples of time dependent mobility. Change in fur color in animals and leaf colors in plants are examples of time dependent thematic variability.

A more detailed overview on the content of the knowledge base could reveal three major levels of change prediction. At a more generalized level, prediction functions can be created based on the existence or non-existence of an object and will be based on the objects behavior in the GIIL. By analyzing the CIIL contents, we can have an estimate on whether to expect that object to change its attributes or not. At the last level, qualitative predictions for change can be introduced. These predictions can be based on an interpolation method applied on the child level. Prediction for new time instances can be extracted through periodicity patterns and more sophisticated graph analysis (e.g. second derivative, cosine functions, etc.). Further analysis of the Knowledge Base component is left for future work.

3.4.3 Change Detection Tools

In the pre-existing model two general components exist, the MIS and the STG. Each one respectively handles implicit and explicit change information. The purpose of this thesis is to provide the essential mechanism, so implicit change is transformed to explicit change. This mechanism cannot be isolated, but should interact with a broader spatiotemporal environment. This is possible now because the framework of this environment is established and change detection can be another component in it. Between the MIS and our proposed DSTG, a new component is introduced, the Change *Detection Tools.*

This set of tools detects changes based on the available information sources and the pre-extracted information in the DSTG. In general, these tools can be categorized as manual, semi-automated, and automated. The manual tools require the (constant) presence of a human operator that would check the available information and extract by hand the changes that might exist. Such a tool could be used within a digitizing environment.

The semi-automated tools combine human and machine input. Several techniques in the image processing field follow this approach. For instance the user is manually picks points, the machine performs some operations based on these points and returns the results back to the user. Such tools have not been utilized for change detection yet, and due to the human factor involved they are not always the best choice in modern environments.

Automated tools are gaining more and more attention the last years. It has become evident that for some tasks it is possible to create fully automated procedures. The lack of human interference in the process makes them more appealing. In addition to that, software and hardware improvements provide the appropriate conditions for this challenge.

Our proposed change detection technique belongs to the last category. It is formulated in such a way that no human interaction is necessary. In order to achieve the human's substitution by a machine, some parts of human knowledge are incorporated in the input through the knowledge base as well as inside the process, as we will see in the following chapter.

Specifically, the Knowledge Base enhances performance and reliability by providing estimations for the expected object change. The MIS provides the essential raster datasets used to extract change. Pre-existing representations of the object are calculated from the DSTG. This information is used in the form of a vector outline. At the output process, when change is detected, the MIS is updated in the case of misregistration errors. Changes are stored in the DSTG, and a procedure is triggered to detect possible new behavior patterns that will be expressed in the Knowledge Base and facilitate future solutions. Based on the presented components and the information flow described, the structure of our extended spatiotemporal environment takes the form shown in figure 3.11. A detailed description of the image analysis approach and its integration with the proposed environment is covered in Chapter 4.

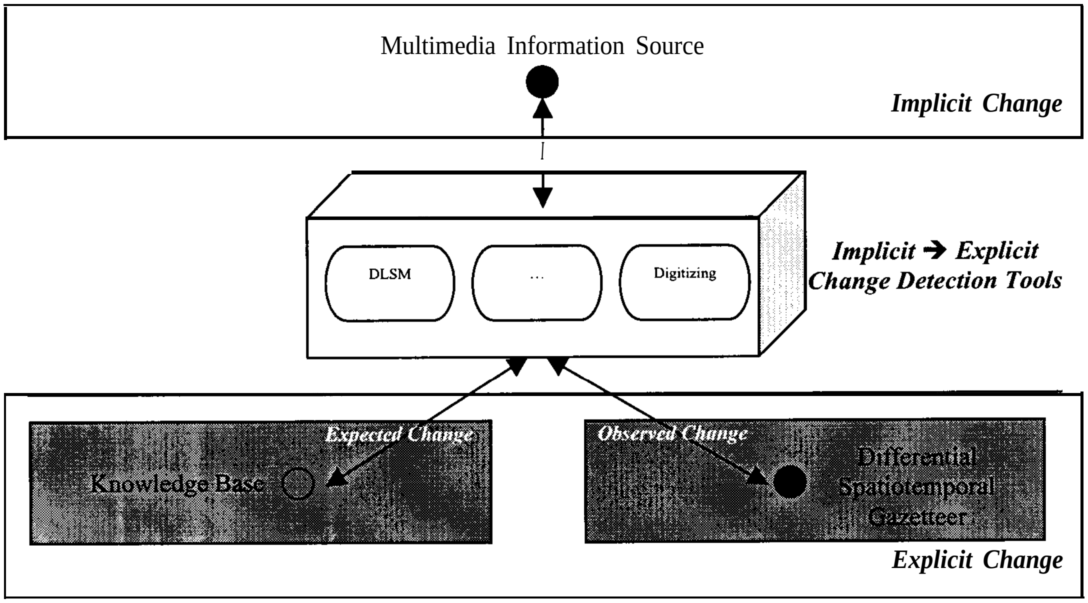


Figure 3.11: Extending DSTG to an Integrated Spatiotemporal Environment to facilitate Change Detection

3.5 Summary

In this chapter a novel spatiotemporal environment for handling change in physical objects is proposed. We use the SpatioTemporal Gazetteer (Beard and Agouris, 1997) as a model that makes more effective use of multiple information resources.

Theoretical analysis is introduced focusing on the choice of reference system for spatiotemporal information, and on change dimensionality and representation issues within a spatiotemporal environment. A change-oriented model is proposed. We save only change information, and more specifically only the dimensions of change that are modified. With this approach we can always ensure minimal redundancy by reducing a multi-dimensional problem to its minimal modified dimensions and maximum efficiency because change is directly available within our environment.

Implementation of this analysis generated the *Differential Spatiotemporal Gazetteer*. This extension of the original gazetteer at a conceptual level is composed of a Geographic Identity Information Level, a Change Indexing Information Level and a collection of child sources the form the Child Information Level. Mathematical relations between these components are presented. The combination of the Geographic Identity Information Level with the other levels provides the ability to query in different change resolutions, from the existence of an object to the detailed aspects of change. Two minimum elements, the Minimum Spatial Element and the Minimum Temporal Element were introduced in the spatial and temporal domains respectively. These elements are user-defined and act as thresholds to avoid the return of unnecessary information. Therefore a significant advantage of this structure is that change is expressed in our environment through the user specifications and needs.

In our proposed model we incorporate the essential tools that extract change. In doing so, the gazetteer is expanded to a *broader spatiotemporal environment* that handles change. Two new components are introduced, the Change Detection Tools and the Knowledge Base. Information flow within the environment is described and component interaction is presented, focusing on the change detection aspects.

Chapter 4

Automated Spatiotemporal Change Detection in Digital Aerial Imagery: Theoretical Issues

Object extraction is a fundamental photogrammetric process. It is the process whereby objects (e.g. buildings, roads) are identified in digital images, and their positional information is recorded. Such information is essential to a large variety of applications, especially GE-related ones. In traditional digital photogrammetric and computer vision approaches, object extraction from digital images comprises two major operations: *identifying* the object within an image, and *precisely trucking* this object by measuring the image coordinates of its outline. The first process involves a variety of logical decisions like image interpretation, understanding, and object classification. The latter is a precise localization problem, aiming at subpixel accuracies in outline detection.

Research in digital photogrammetry and computer vision during the last 20 years showed that there exist no universal edge detectors which can be applied to a digital image function to both identify and track edges with sufficient success. Instead, there exists a trade-off between:

- *reliability,* which expresses the qualitative accuracy associated with identification, and
- *precision,* which expresses the geometric accuracy associated with tracking.

Operators that excel in reliability are often referred to as *type I operators, in* analogy to type I errors in statistics. They succeed in identifying classes of objects (e.g. houses) within an image, without particularly dealing with precise outline localization. Their products may be for example object blobs. Operators that excel in accurately positioning object outlines are often termed *type II operators.* Typically, they function successfully within narrow search windows. Operators from these two broad classes have often been combined, in an effort to optimize both accuracy measures (Suetens et al., 1992). A good overview in man-made object extraction from digital imagery, in both digital photogrammetry and computer vision may be found in (Gruen et al., 1997; Gruen et al., 1995).

A rather surprising trend in object extraction algorithm development is that the majority of approaches treat geospatial information collection as a static process. Accordingly, they formulate the problem as that of extracting an outline from an image, regardless of prior information that might exist for the depicted area in general and the specific object in particular. Considering today's practice trends, this static view tends to become increasingly obsolete. Instead, object extraction becomes a crucial part of larger cycles of GIS updates.

In this chapter we present our approach for the identification of changes in object outlines within a broader spatiotemporal environment. This approach is based on the use of least squares template matching, where prior data are analyzed to provide template information. The product of such a process is the identification of changes in object outlines and the subsequent update of a GIS to ensure its continuous validity. In this approach, pre-existing information plays the role of a type I operator, providing the necessary approximations for our template matching tools. Least squares template matching acts as a type II operator, used to extract precise outline information from a digital image. This work is innovative in:

- its use of prior information to provide templates for matching, and
- its analysis of template information to assign proper weights in the least squares solution.

This chapter presents the theoretical models behind our approach for automated change detection. In the next section we present an overview of this approach and its interaction with the spatiotemporal model and the query method. A description of object decomposition to its primitives follow in the next section. Section 3 presents the differential least squares matching technique and a reasoning process for distinguishing actual changes from photogrammetric systematic errors is presented in Section 4.

4.1 Approach Overview

In modern geospatial applications object extraction becomes increasingly part of larger cycles of GIS updates. Accordingly, the objective is to compare new information to older versions, and to identify changes that have occurred in the meantime. In our approach, this is equivalent to comparing an object as it is represented in an image captured at instance *T* to the same object recorded in a GIS at *T-dt*. This is a matching problem where an object is compared to itself in order to identify if it has changed.

4.1.1 Query Interaction

The change detection process can be triggered automatically when new information arrives in the MIS (Multimedia Information Store). This would update our change environment based on the content of the new image. Another approach would be to establish a connection between the user and this tool, so that specific spatiotemporal change can be obtained. As shown in chapter 3, during the query process the user has the flexibility to define:

- the spatial resolution through the MSE (Minimum Spatial Element) function
- the temporal resolution through the MTE (Minimum Temporal Element) function
- the spatial extent, and
- the temporal extent of the query.

Based on the spatial and temporal extent the available sources are retrieved. They should have a spatial representation of the object(s) of interest in that temporal interval. We say objects and not area, because as analyzed in chapter 3, we model the spatiotemporal space through changes in discrete objects. The use of temporal interval instead of temporal point is justified by the nature of change. It is expressed as a difference between two states that in real world can only take place in a time interval, no matter how small that is.

Then we make use of the temporal resolution (MTE) to filter unwanted information. It is impossible to achieve continuous temporal representation through imagery, so the time interval has to be projected to an aggregation of time points. Based on the MTE, these temporal points are extracted and the information sources are refined. The final step to create our working dataset is to examine the spatial resolution of the images. This is performed with the use of the MSE which is pre-defined by the user. Later on we will see how the MSE is establishing the matching points for our technique.

Here we should note that from the levels of change that were identified in Chapter 3, namely Existence of Object, Existence of Object Change, and Characteristics of Object Change, the image analysis technique proposed in this thesis addresses queries of the second level (Existence of Object Change). Specifically, our technique examines whether an outline of an object has changed or not.

4.1.2 Model Interaction

After the datasets are chosen, the information retrieval from the model can begin. This interaction of our change detection process with the other components of our model is shown in figure 4.1. From the DSTG (Differential Spatiotemporal Gazetteer), old object vector information from time *T-dt* is retrieved. Following this, the corresponding to the extracted vector georeferenced images are retrieved by using positional information of the vector. These images are retrieved from the MIS and correspond to time *T* and *T-dt.* Another type of input for the algorithm, a knowledge one, is inserted. It expresses the elasticity of the object as presented in chapter 3. Note that the content of the knowledge base is not fully addressed in this thesis. We are focusing on how certain specific pieces of information may be introduced in a pre-existing knowledge base and how they interact within the change detection process. So how the elasticity is obtained is not addressed but the incorporation of such available information is of interest. The latter is performed by adjusting accordingly the window size of the pattern. This size is also adjusted based on accuracy estimations either in the image domain coming from the MIS (e.g. georeferencing accuracy), or the vector one (e.g. extraction accuracy) provided by the DSTG.

Following the matching process, the obtained results are filtered through a reasoning operation. In doing so, we distinguish actual change from systematic sensor errors. Different directional patterns and models expressing these errors are imported from the knowledge base, depending on the particular image (e.g. misregistration) and sensor (e.g. radial distortion) type. If such errors are detected, the information level is updated by correcting the metadata of the used source. When no change is detected, then that information is stored as well by accessing the appropriate Information Levels, the Geographic Identity Information Level (GIIL) to verify the existence (or not) of the object(s) and the Change Indexing Information Level (CIIL) that provides an index for the detailed attributes of change. This is happening because that is still change information for our model. So no matter what the filtering result is, change dimensions are stored in the appropriate components of the DSTG. If the object no longer exists then the GIIL is updated. If it does exist child source(s) are accessed, and depending on the dimensions that are modified the CIIL is updated too. At the same time, a verification of the object's behavior model could be performed, and if it differs from the one already stored, the knowledge base is updated as well. A more comprehensive description for the interaction of the change detection tool with the spatiotemporal model can be found in (Agouris et al., 2000c).

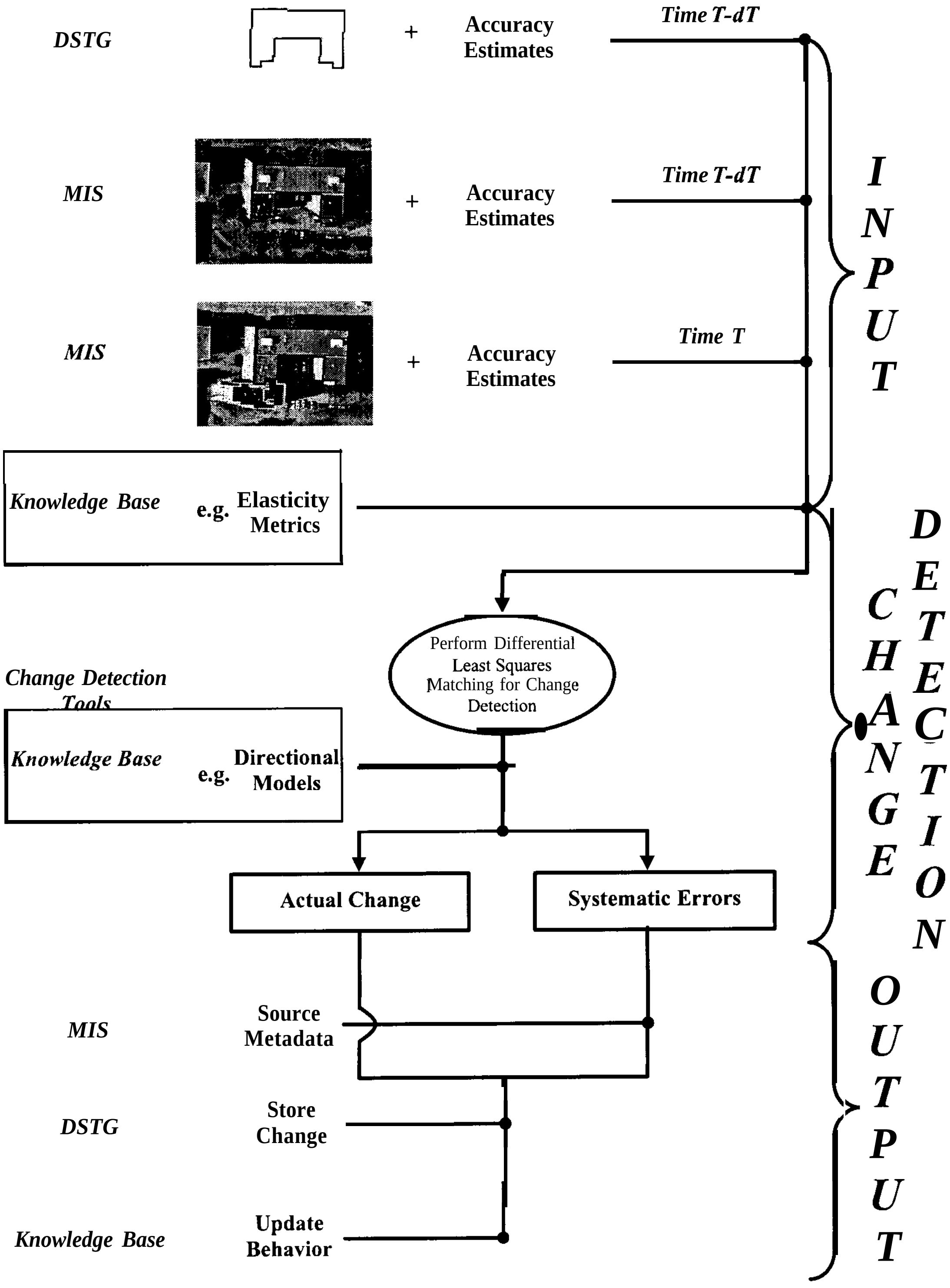


Figure 4.1: Information Flow in the Spatiotemporal Model for Change Detection

4.2 Object Decomposition

In the development of concepts for automatic building extraction, object decomposition is a major issue. A variety of solutions have been proposed for various types of input data (e.g. DEM or raster images). Modeling schemes can be classified into *polyhedral models*, *prismatic models* and *parameterized volumetric primitives* (Fisher et al., 1999). The nature of our application allows prior representation of an object in both the raster domain (e.g. radiometric values) as well as the vector domain (e.g. outline vector data). Since vector information is already available in the form of older GIS data, we do not have to follow a hypothesize-and-verify procedure.

For the purposes of our approach, a 3D object is expressed through a wireframe representation (de Cambray, 1993; Haala and Brenner, 1999; Vosselman and Veldhuis, 1999) using prior vector information. Under the assumption that the employment of this algorithm is focused on change detection for building outlines, the examined object may be considered as an aggregation of planar surfaces, following the concept of polyhedral models (Bignone et al., 1996; Hendrickx et al., 1997). A generalization of planar surfaces may be performed by assuming that they are equivalently represented by intersections of planes, i.e. by lines. Due to the nature of our raster datasets (aerial photography), vertical aerial photos are assumed to be available, and the 3D planes are merged into 2D by extracting a 2D projection from our 3D vector based on the image viewpoint position.

For further decomposition of the lines, the Minimum Spatial Element (MSE) (Mountrakis et al., 2000), introduced in Chapter 3, is used. The MSE describes the resolution of spatial change that the user is interested in. We can use absolute estimates for this resolution (e.g. 2m on the ground, or 0.5 pixel on the image), or relative measures (e.g. the average size of a room). According to the application, MSE estimates can be preset to default values, or they can be dynamically updated by the user.

Regardless of these choices, eventually we obtain an absolute value which defines change resolution for the specific application. Using this information, we perform a segmentation of outlines, and lines are essentially substituted by the corresponding points along the outline (Fig. 4.2). As corners are defined by line intersections, we do not have to consider them in our outline decomposition. A final product of this process is moving from a 3-D object to a set of points in the 2-D image space.

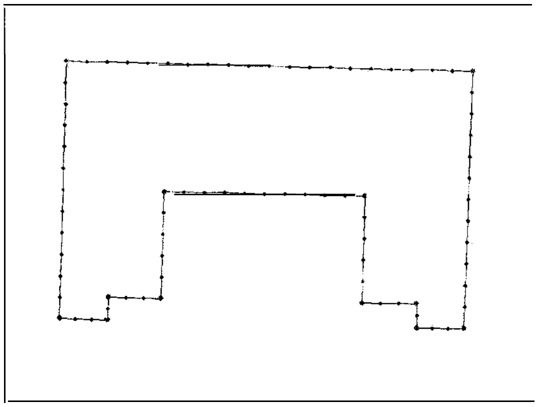


Figure 4.2: Object Outline Decomposition

4.3 Differential Least Squares Matching

4.3.1 Algorithm Workflow

In order to detect changes on the outline of a building we compare an object as it is represented in an image captured at instance *T* to the same object at *T-dt*. First, the older object vector information is retrieved from the corresponding child GIS source (Fig. 4.3). After performing object decomposition on this vector information, as described in section 4.2, we choose candidate matching points. These are the points where our differential application of least squares template matching (DLSM) takes place to detect change.

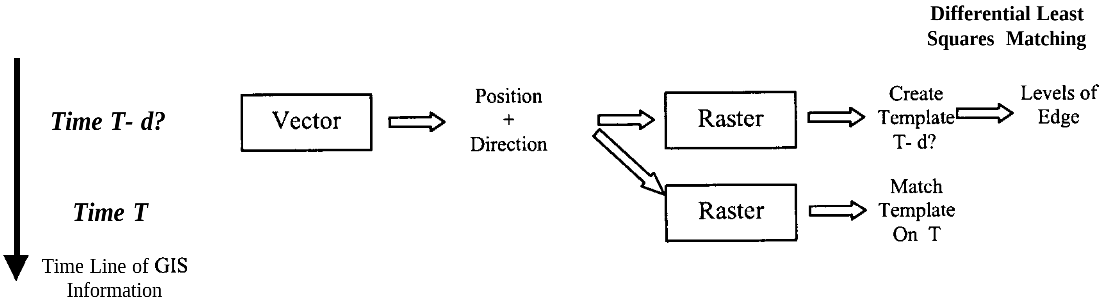


Figure 4.3: Process Flowchart for Outline Change Detection

Analysis of the object outline provides us with local edge position and orientation information to facilitate computations, as we will see in the following section. Elasticity metrics and vector accuracy estimates are incorporated in the matching process by using variable window sizes (large windows for low accuracy/highly elastic data and vice versa). We optimize computational performance without compromising accuracy potential by choosing the smallest applicable window size.

By matching an edge template to the image we identify outline locations in this image as conjugates to template edge locations. Thus we transfer the high accuracy potential of least squares matching (LSM) onto object extraction. By performing this process at select positions along the object outline we compare the complete object to its last recorded information. Changes are identified as instances where DLSM fails to match adequately (Agouris et al., 2000a).

In order to visualize our approach let's consider two raster images from different times (Fig. 4.4). In this figure we show two conjugate windows that will be used as input for a differential application of least squares template matching.

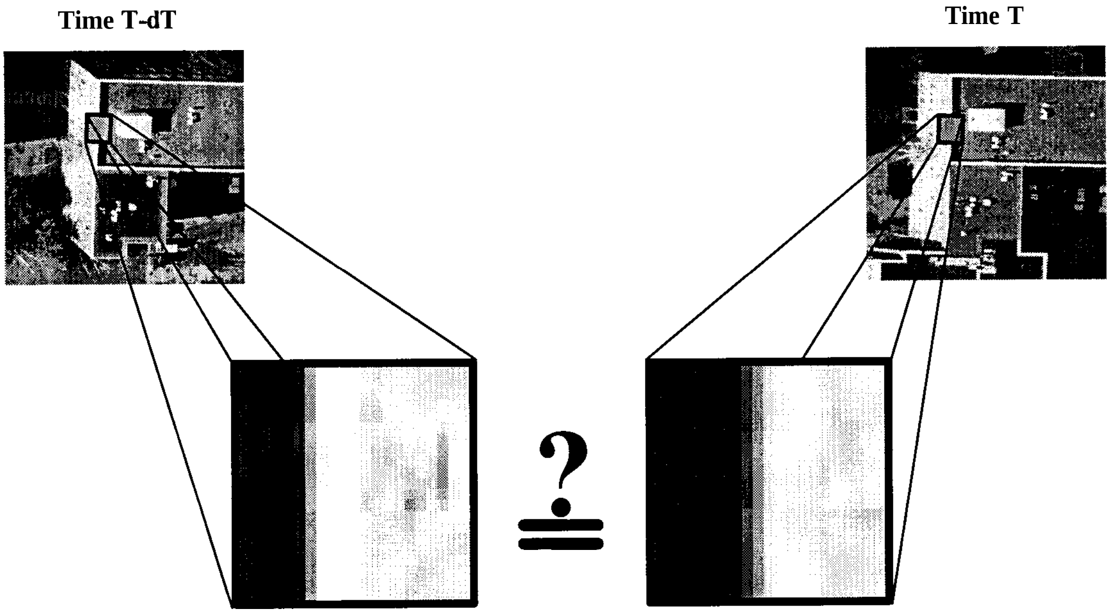


Figure 4.4: Hypotheses of our Matching

In order to take advantage of the preexisting raster information and to enhance the algorithm's performance and liability, a geometric representation of the edge is extracted and included in the DLSM solution by modifying accordingly the weight matrix elements. Several criteria are taken under consideration for a successful match, such as:

- values of the parameters,
- error estimates for these parameters, and
- estimates of the overall performance of our model.

By matching an edge template to the image we identify outline locations in this image as conjugates to template edge locations. Changes are identified as instances where DLSM fails to match adequately. Information from this process is used to update our spatiotemporal environment.

4.3.2 Mathematical Model

The method we are presenting here employs least squares matching for the detection and tracking of edges in digital images (Gruen, 1985; Gruen and Agouris, 1994; Gruen and Stallmann, 1993). A window depicting an edge pattern is introduced as a reference template that is subsequently matched to digital image patches in the vicinity of actual edge segments. The concept behind the method is simple yet effective: by matching the edge template window to an image window, we can identify edge locations in the image as conjugate to the a priori known template edge positions.

Assuming *f(x,y)* to be the reference edge template and *g(x,y)* to be the actual image patch, a matching correspondence is established between them when

$$f(x,y) = g(x,y) \tag{4.1}$$

However, considering the effects of noise in the actual image, the above equation becomes

$$f(x,y)-g(x,y)=e(x,y) \tag{4.2}$$

with *e(x,y)* being the error vector.

In a typical least squares matching method, observation equations can be formed relating the gray values of corresponding pixels, and they are linearized as

$$f(x,y)-e(x,y)=g^{o}(x,y)+\frac{\partial g^{o}(x,y)}{\partial x}dx+\frac{\partial g^{o}(x,y)}{\partial y}dy \tag{4.3}$$

The derivatives of the image function in this equation express the rate of change of gray values along the x and y directions, evaluated at window pixels. Depending on the type of edge, the geometric relationship describing the two windows may be as complex as an affine transformation, or as simple as a simple shift and/or rotation. To facilitate solutions we can resample template and/or actual image to have edges lying along one of the major coordinate axes. A set of a shift and a rotation is incorporated in our model. In this case we can have resampled edges oriented along the y axis of the corresponding windows. Then equation (4.3) may be reduced to:

$$f(x,y)-e(x,y)=g^{o}(x,y)+\frac{\partial g^{o}(x,y)}{\partial x}dx \tag{4.4}$$

The two patches are geometrically related through a shift along the x direction that expresses the combining result of a shift and a rotation:

$$X_N = X_o + dX \tag{4.5}$$

The dX parameter is the unknown that allows the repositioning of the image window to a location that displays better radiometric resemblance to the reference template. It is introduced in the derivative terms $(\partial g/\partial x)$ of the linearized observations above as

$$f(x,y)-e(x,y)=g^{o}(x,y)+g_x d\delta X \tag{4.6}$$

Regardless of the choice of geometric transformation, the resulting observation equations are grouped in matrix form as

$$- e = A x - 1 \quad ; \quad P \tag{4.7}$$

In this system, *l* is the observation vector, containing gray value differences of conjugate pixels. The vector of unknowns x comprises the shift at the x direction, while A is the corresponding design matrix containing the derivatives of the observation equations with respect to the parameters, and ***P*** is the weight matrix. A least squares solution allows the determination of the unknown parameters as

$$\hat{x} = (A^T PA)^{-1} A^T Pl \tag{4.8}$$

Based on the formulation of the above model, we have to face two major challenges,

- the formulation of the weight matrix, and
- the evaluation of the obtained results.

4.3.3 Geometry of the Edge

4.3.3.1 Levels of the Edge

In order to take advantage of the raster representation that preexists and enhance the algorithm's performance and reliability, a geometric representation of the edge is extracted and included in the DLSM solution (Agouris et al., 2000b). Depending on the edge type, multiple levels can be extracted based on the edge's radiometric representation. Since change detection in buildings is the focus of our work, we categorize edges as having two, three or four levels, depending on the building's edge representation (Fig. 4.5). On the image this corresponds to one, two or three edges (two levels form an edge) and in reality to specific man-made structures, like curbs at the edges of buildings.

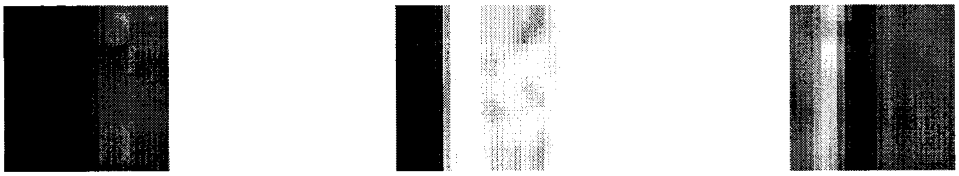


Figure 4.5: Two, Three and Four Levels of Edge Representation

The window is resampled so that the edge is perpendicular to the x-axis of the window under examination. Then we calculate the average first derivative of each column along the X direction:

$$g_x(x,y) = \frac{\partial g(x,y)}{\partial x} \tag{4.9}$$

$$\bar{g}_x(x) = \frac{\sum_{1}^{y} g_x(x,y)}{Y} \tag{4.10}$$

Clearly, the derivatives along the y-axis will be minimal for our set-up. To compensate for noise, a subset of 80% of the pixels with the lower residuals is used. Analysis of the $\bar{g}_x$ (x) graph (Fig. 4.6) using the first and second derivatives reveals 3 (or less) maxima.

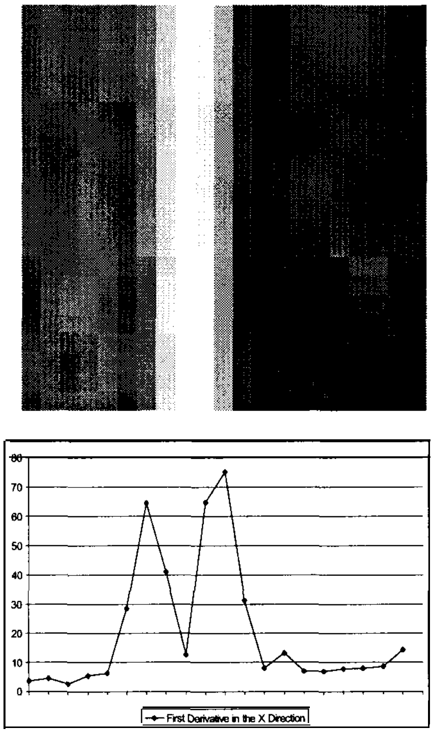


Figure 4.6: Template Window (top) and Corresponding Averages of the First Derivative in the X Direction (bottom)

After the extraction of these peaks from the graph (max_1,max_2,max_3), a filtering process follows to verify if it corresponds to an image edge or not. To normalize the results within the filtering, the maximum peak is extracted and used within these criteria (max_o):

$$max_0 = Max\{ maxi , max_2 , max_3 \} \tag{4.11}$$

Three criteria are introduced in this decision process. The first one attempts to detect an edge based on its strong presence, relative to the maximum peak. This provides a fast way of accepting a peak.

If $maxi > C_1 * max_o$ then accept peak.

The constant C_1 expresses a threshold, beyond which the relative sharpness reveals a strong edge.

Another used factor is the inverse of the above. If a peak has a weak presence compared to the maximum peak, then it should not be further processed. With this criterion peaks are rejected.

If maxi $< C_2 * \max_o$ then reject peak.

The constant C_2 expresses a threshold, below which the relative sharpness reveals a weak edge.

Finally, for the more complicated case, where a peak presence cannot be classified as weak or strong, a more detailed analysis is performed. We expand the criterion to include neighbor points to this peak, so a more thorough estimation can be performed. We calculate the rate of change (slope) by comparing the peak value with the previous and the next one, and if a sharp (sudden) peak is detected, then it is accepted, otherwise it is rejected. This is expressed through the constant C_3.

If $C_2 * \max_o <$ maxi $< C_1 * \max_o$ then

$$\left|\bar{\nabla}_x(x)\right| + \left|\bar{\nabla}_x(x+1)\right| \leq C_3 \max_o \tag{4.12}$$

where x is the coordinate along the axis perpendicular to the edge, $\max_0$ is the maximum of the three maxima found, and $\bar{\nabla}_x$ is the average second derivative function.

With the three above criteria, a ‘relative” check based on maximum value compares the dominant edge with other gradient maxima to eliminate false responses. Before we finalize the levels we perform a last check by introducing the first derivative in the Y direction. We claim that an edge should have high gray value variations perpendicular to itself but at the same time show low variations along itself. In this step, we check for consistency along the Y-direction, but only where the edges (i.e. accepted maxima) that came from the previous step are not strong enough. The difference from the criteria used before is that this time absolute gray values are used, acting as thresholds, and not relative variations of gray values. During this process two thresholds are introduced:

- the *DX-threshold*, showing the difference in the gray values along the X axis that might not imply an edge, and
- the *DY-threshold*, expressing the difference in the gray values along the Y axis that might not imply homogeneity.

The DX-threshold is applied on the accepted peaks of the previous step. If the condition

$$\text{maxi} < \text{DX-threshold} \tag{4.13}$$

is satisfied, there is the suspicion that local noise might exist or the radiometric representation of the edge is not strong. To distinguish these two, we make use of the DY-threshold. We compute the first derivative in the Y direction at the position where the candidate maximum is, and if it is smaller than the DY-threshold, then that point is accepted.

4.3.3.2 Expressing Geometry Semantics through the Weight Matrix

In the previous section, analysis on prior raster information provided a description of the geometry of the edge that we are trying to match. This geometric information is inserted in our mathematical model by formulating accordingly the weight matrix. The problem we had to face was how to express the levels of the edge by a mathematical function that formulates the weights accordingly. We choose to use the Gaussian Distribution, because:

i. It is a continuous distribution. Our observations might be discrete (digital imagery), but we try to model the continuous reality.

ii. It is a symmetrical distribution. With this we can expect the same weight distribution in both sides of the edge.

iii. It is a probability distribution. The matrix formulation is based on the probability of a pixel value to represent an edge or not.

iv. Many naturally occurring noise sources can be described by it (e.g. white noise). This justifies our choice, because noise is a major problem that we try to compensate for.

v. It is a well-known and broadly established mathematical distribution.

The mathematical expression of the Gaussian Distribution (GD) is shown below.

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{(x-\mu)^2}{-2\sigma^2}} \qquad (4.14)$$

In this formula μ is the mean, s is the standard deviation and x is the coordinate of the pixel on the axis perpendicular to the edge. If we apply the GD in a template with one edge, then the mean expresses the position of the edge as defined by the older vector information. The standard deviation defines the uncertainty for the existence of the edge on the position of the mean. The standard deviation depends on the resolution of an image, because in higher resolutions the edge is expected to look ‘Sharper”, and in coarser resolutions more “blurry”. In addition to that the accuracy of the vector data can be incorporated in the solution. Higher standard deviation means more spread distributions that takes into account more pixel values. With this low confidence on the extracted vector can be expressed.

The formulation of the GD of the weights in the case of an edge represented by two levels (left image in Fig. 4.5) is straightforward. In many cases though the edge can be represented by three or more levels (center and right image in Fig. 4.5). Then a more complex formulation of the weights takes place. In the previous section (4.3.3.1) the process for identifying possible multiple levels of the edge representation was explained. In this section we make use of the positions of these extracted levels.

As we saw earlier the levels are formed through a combination of accepted maxima on the first derivative perpendicular to the vector edge. A combination of Gaussian Distributions can be applied in order to assign higher weights at these accepted maxima. For the purposes of our modeling, let's assume that an edge is composed of three levels (i.e. two edges). The original vector position is μ_2 and the other accepted maximum is on position μ_1 (Fig. 4.7). Here we should note that μ's represent distances vertical to the edge (i.e. along the X axis). We define as d the distance between the two μ. Two GDs are used with their means μ_1 and μ_2, respectively. The mathematical representations for these distributions are:

$$G_1(x) = e^{\frac{(x-\mu_1)^2}{-2\sigma_1^2}} \text{ , } G_2(x) = e^{\frac{(x-\mu_2)^2}{-2\sigma_2^2}} \tag{4.15}$$

Note that the $\frac{1}{\sqrt{2\pi\sigma}_i}$ factor is omitted. This factor is a scaling constant of each GD. By not including this factor, both of the GDs are normalized between zero and one. In order to formulate the GDs and in essence the weight matrix, we need to define the two values for the standard deviations s_1 and s_2.

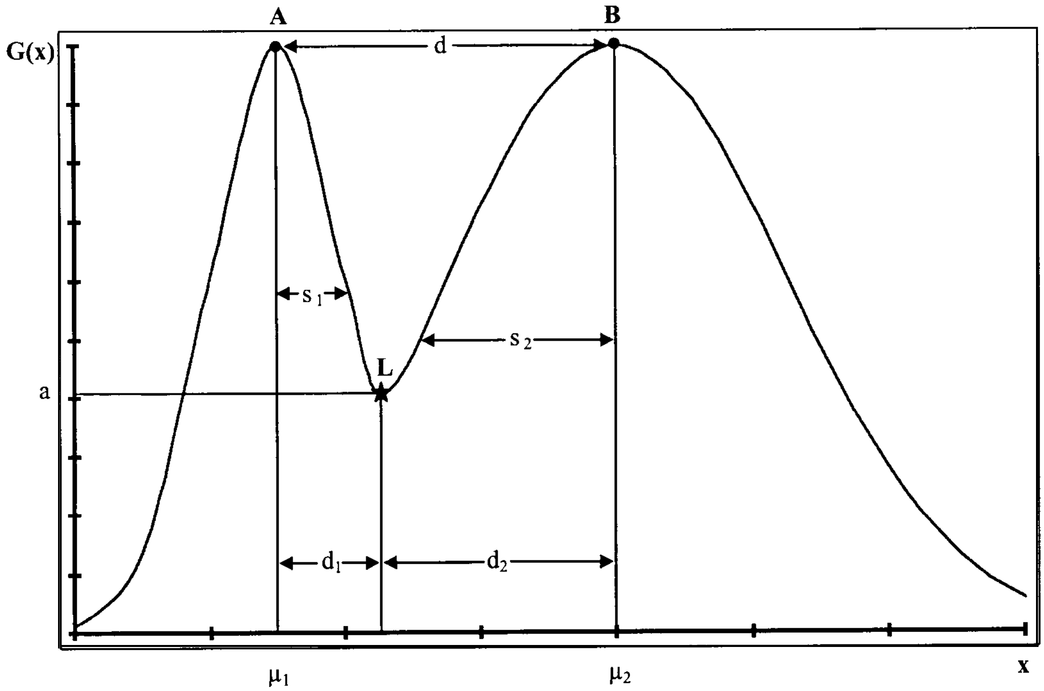


Figure 4.7: Calculation of Standard Deviations

We are solving for two parameters (s_1, s_2) so we need two equations. The first one is extracted from the fact a point L should exist, where the two GDs intersect. Let d_1 and d_2 be the two distances from μ_1 and μ_2, respectively and ***a*** is the GD value of point L. The first equation is calculated as follows:

$$\left.\begin{array}{l} G_1(\mu_1 + d_1) = G_2(\mu_1 + d_1) = a \\[2ex] G_2(x) = e^{\frac{(x-\mu_2)^2}{-2\sigma_2^2}},\ G_1(x) = e^{\frac{(x-\mu_1)^2}{-2\sigma_1^2}} \\[2ex] x = \mu_1 + d_1 = \mu_2 - d_2 \end{array}\right\} \left.\begin{array}{l} \sigma_1 = \dfrac{d_1}{\sqrt{-2\ln a}} \\[3ex] \sigma_2 = \dfrac{d_2}{\sqrt{-2\ln a}} \end{array}\right\} \begin{array}{l} \sigma_1 + \sigma_2 = \dfrac{d_1 + d_2}{\sqrt{-2\ln a}} \\[3ex] d_1 + d_2 = d \end{array}$$

$$\sigma_1 + \sigma_2 = \frac{d}{\sqrt{-2\ln a}} \tag{4.16}$$

In order to compute the standard deviations of the gaussians another constraint should be introduced. Due to the fact that point B is a pre-extracted edge and point A is a computed peak, we do not want point A to influence the solution beyond point B. That can be statistically ensured by restricting the standard deviation. For a threshold of 99% for the possibility of the observations to fall inside an interval, the standard deviation should be three times that interval. We make use of this and another equation based on the above is introduced, showing this relation. It is expressed as:

$$3s_1 = d \tag{4.17}$$

We substitute that in the first equation and we calculate s_2.

$$\sigma_2 = d\frac{3 - \sqrt{-2\ln a}}{3\sqrt{-2\ln a}} \tag{4.18}$$

Because standard deviation is always positive, the value of a should always satisfy the equation:

$$a \geq e^{-4.5} \tag{4.19}$$

The ratio of s_2/d as the multi-scale factor a changes from $e^{-4.5}$ to 1 is shown in figure 4.8.

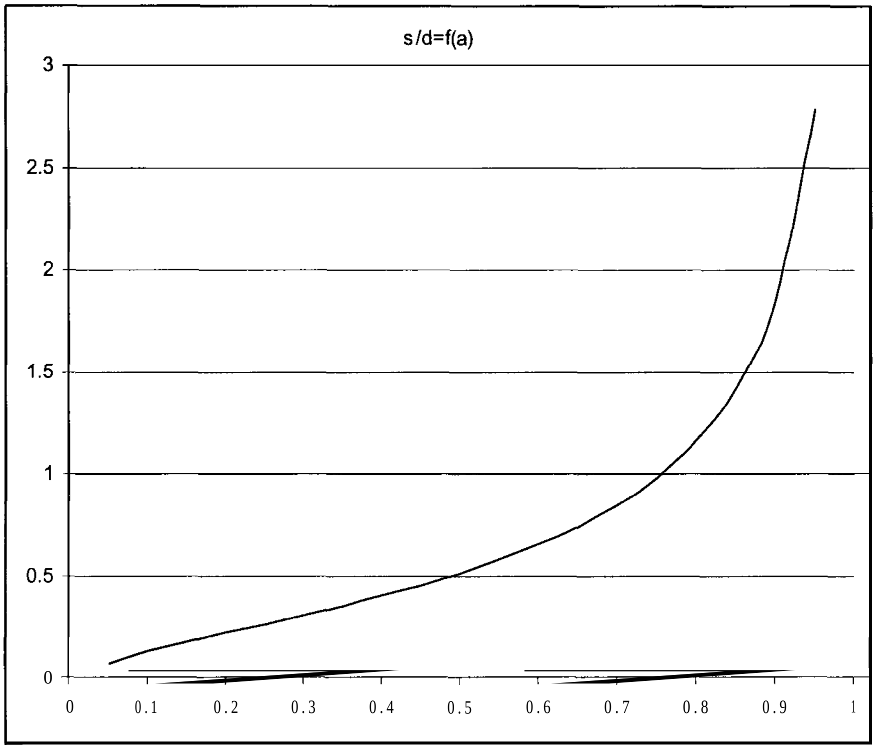


Figure 4.8: Changes in the s/d Ratio based on the a Multi-scale Constant

Another restriction is applied in this process to compensate for the fact that the original edge we try to match (vector edge) corresponds to s_2. Pixels around this edge (μ_2) should contribute more to the solution, or at least as much as the pixels around the calculated edge (pi). Mathematically this is ensured if:

$$s_2 > s_1 \tag{4.20}$$

If we combine the equations (4.17), (4.18), (4.20) we find:

$$a > 0.32 \tag{4.2 1}$$

The a constant provides the ability to formulate the GDs based on a scale-space factor. As we previously mentioned, the same edge can have different representations as a result of different image resolutions. The edges are more distinct at higher resolutions, and more blurry in lower resolutions. Constant a expresses the weight of the intersecting point. If that weight is low, then the standard deviation of the second GD will be small, and the weights will be more dense near the point B. This is applicable in the case of higher resolution, where the edges are distinct. On the other hand, if we choose a high intersecting value, then the GDs get wider and increase the contribution of surrounding pixels to point B. This is desirable in low resolution imagery, where edges are not so distinctive. The graph below (Fig. 4.9) shows the dependency of the second GD distribution on the intersecting value a (y axis).

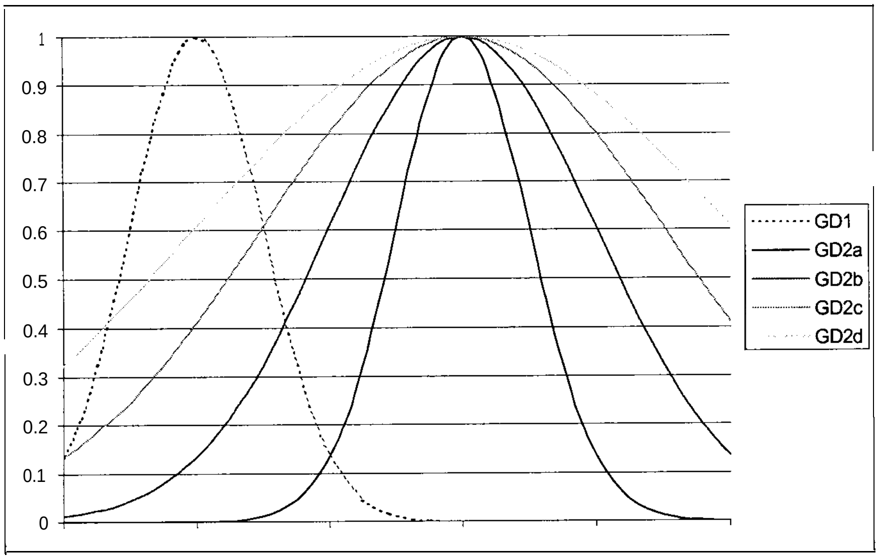


Figure 4.9: Scale Dependent Formulation of Weight Matrix

An example of the formulation of the weight matrix depending on the number of levels in shown in figure 4.10.

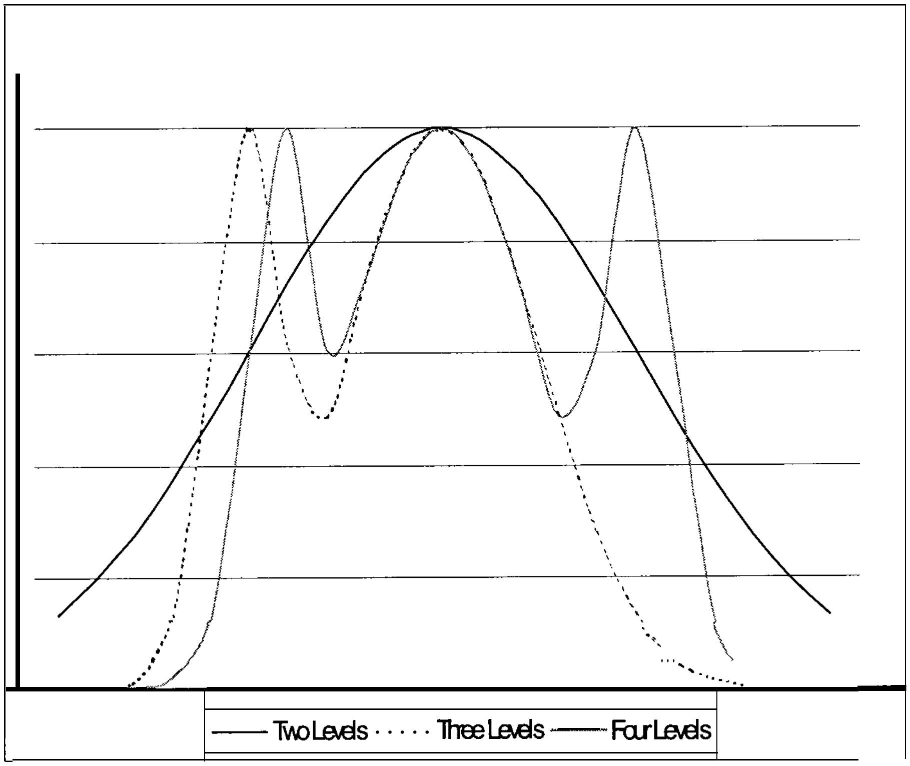


Figure 4.10: Gaussian Distribution of Weights

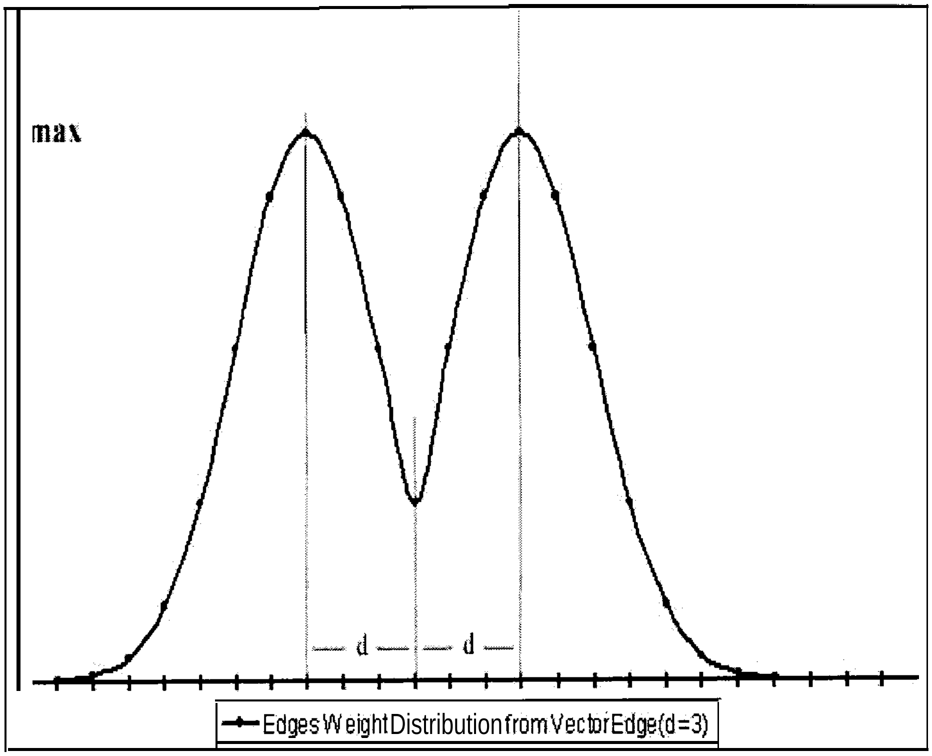


Figure 4.11: Gaussian Distribution of Weight Scales based on Expected Geometry

- **Expected Geometry**

When multiple edges exist, they can result from the actual shape of the edge or random conditions such as noise, shadows, different projections, etc. We compensate for this by introducing a *scaling factor* describing the *expected geometry* of the edge. This is applicable in cases where curbs on the edges of buildings exist. Two new mirror GDs are created on both sides of the vector edge, based on the actual width of the expected curb (*d*) (Fig 4.11). The means of these GDs are assigned the values of $\mu_{vector} \pm d$, depending on which side they are. The standard deviation of the expected geometry GDs expresses the uncertainty of this ‘knowledge” information and it is defined as a constant based on the distance *d*.

The mathematical expressions of the two mirror GDs $K_i(x)$ are:

$$K_i(x) = \mathrm{Max}\{e^{\frac{(x-\mu_{vector}-d)^2}{-2\sigma_i^2}},\ e^{\frac{(x-\mu_{vector}+d)^2}{-2\sigma_i^2}}\} \tag{4.22}$$

Note again that the factor $\frac{1}{\sqrt{2\pi\sigma}_i}$ is omitted to normalize these scaling factors from zero to one. What we achieved with this method is that pixels close to the expected distance *d* will have almost the same weight as the vector edge, while pixels away from that will be scaled down. In the evaluation part of this thesis (Chapter 5) the significance of this analysis is presented. An example of the incorporation of the expected geometry can be seen in figure 4.12.

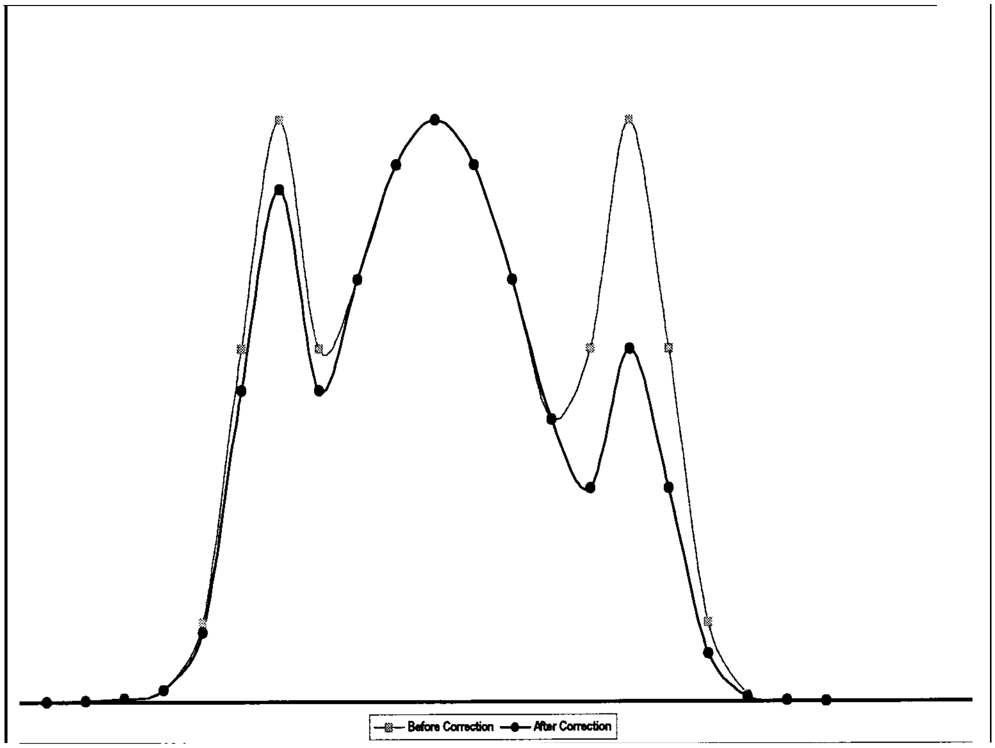


Figure 4.12: Gaussian Distribution Before and After the Expected Geometry Correction

- **Formulation of weight matrix**

The weight distribution P(x) on the axes perpendicular to the vector edge is given by the formula:

$$P(x) = \text{Max}\,\{K_1(x)*G_1(x), G_2(x), K_3(x)*G_3(x)\} \qquad (4.23)$$

$$\text{with } K_i(x) = \text{Max}\{e^{\frac{(x-\mu_{vector}-d)^2}{-2\sigma_i^2}}, e^{\frac{(x-\mu_{vector}+d)^2}{-2\sigma_i^2}}\}, G_i(x) = e^{\frac{(x-\mu_i)^2}{-2\sigma_i^2}} \qquad (4.24)$$

The GDs K_i express the expected geometry scaling factors and the GDs G_i show the distributions formulated after incorporating the edge representation analysis in the process. Note that the equation 4.23 corresponds to figure 4.12, where three peaks were identified from the edge representation and the vector edge is the middle one. For a different setup the weight distribution is formulated accordingly. For example if the vector edge would be the first one, then the weights would be expressed as:

$$P(x) = \mathrm{Max}\,\{G_1(x), K_2(x)*G_2(x), K_3(x)*G_3(x)\} \qquad \textbf{(4.25)}$$

Due to the fact that weights contribute to the calculation of the standard deviation s?, an equal projection should be established so the s ?'s from different weight distributions result in the same contribution. To incorporate that, the sum of the weights is defined to be 1 by using the formula:

$$P(x) = \frac{P(x)}{\sum P(x)} \qquad \textbf{(4.26)}$$

At this point we should note that for the first three iterations of the least squares solution, all weights are scaled to a 0.5 to 1 range instead of 0 to 1, so that the whole window is contributing to the solution, and possible edges near the sides of the window are not excluded from the solution.

4.3.4 Decision Support

The statistical tools of least squares adjustments provide the mathematical foundation necessary to perform analysis of the obtained results, and to automatically decide whether or not change occurred. From the implementation view, our algorithm is based on a dynamic programming approach. Iterations are performed and based on the collected results, a decision should be made, whether they are satisfying and the process should be terminated, or whether the process should continue. During this execution of the matching loop several criteria are considered to support this decision.

4.3.4.1 Shift dX

As we described in the mathematical model, we relate geometrically the window and the image through a shift dX. This shift is calculated from the below equation:

$$\hat{x} = (A^T P A)^{-1} A^T P l \qquad (4.27)$$

where A is the corresponding design matrix containing the derivatives of observation equations with respect to parameters, *P* is the weight matrix and *l* is the observation matrix. Depending on the value of this shift a decision can be made whether the desired pixel accuracy has been reached and if another iteration should follow.

4.3.4.2 A Posteriori Variance of Unit Weight

Another statistical tool for the analysis of adjustment results is the a posteriori variance of unit weight. This variance expresses how well the overall model fits the observations. It is computed by the following formula:

$$\hat{\sigma}_o^2 = \frac{V^T P V}{r} \qquad (4.28)$$

where *V* is the matrix containing the residuals, *P* the weight matrix and ***r*** the degree of freedom.

4.3.4.3 Variance-Covariance Matrix for Parameters

This matrix shows a statistical estimation for the accuracy of the obtained results. It is computed as:

$$\sum_{\hat{x}} = \hat{\sigma}_o^2 (A^T P A)^{-1} \qquad (4.29)$$

where A is the corresponding design matrix containing the derivatives of observation equations with respect to parameters, ***P*** the weight matrix and s $_o$ the a posteriori variance of unit weight. The information contained therein describes the error ellipses of the estimated parameters.

4.4 Distinguishing Actual Change from Sensor Errors

After DLSM, new positions for edges are established. As a post-process a critical reasoning for our application is performed to distinguish actual changes from deviations inherited from systematic errors of the source (e.g. misregistration, internal orientation errors). We compare change residuals with directional patterns that describe specific deformations in aerial photogrammetry. Well-established photogrammetric mathematical models (e.g. exterior and interior orientation) that describe these deviations are interpolated between the prior and the new edge positions. A least squares solution provides confidence for the success of the fitting models, and if such models are detected, the relevant metadata in the Multimedia Information Store are updated to help future solutions. Here we should note that in order to obtain accurate results a well-distributed set of points on the image is required.

Let's assume that we have two sets of matching points from two different time instances; the new object coordinates (X_N , Y_N), and the old object coordinates (Xo , Yo). The new set of points describing the new object boundary has one-to-one correspondence with the old set, since it came as a result of successful matching.

4.4.1 Exterior Orientation Corrections

In this case we examine the change residuals and we compare them with directional patterns (Fig. 4.14) that describe exterior orientation errors in vertical aerial images. An exterior orientation of an image solves for the following parameters (Fig. 4.13):

- Translation along axis X (? X)
- Translation along axis Y (? Y)
- Translation along axis Z (? Z)
- Rotation around the X axis (? ?)
- Rotation around the Y axis (? f)
- Rotation around the Z axis (? ?)

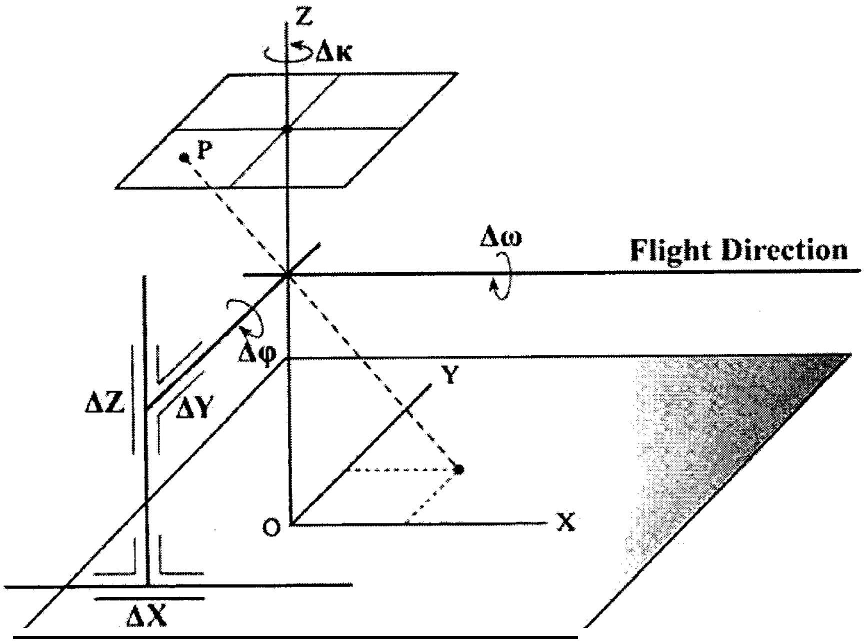


Figure 4.13 : Exterior Orientation Parameters of a Single Photo

In order to formulate our mathematical model, we use the colinearity equations that relate ground coordinates with image ones. Accordingly, our observation equations are:

$$X_0 - X_N = c\frac{\delta X * R_{11} + \delta Y * R_{12} + \delta Z * R_{13}}{\delta X * R_{31} + \delta Y * R_{32} + \delta Z * R_{33}} \qquad (4.30)$$

$$Y_0 - Y_N = c\frac{\delta X * R_{21} + \delta Y * R_{22} + \delta Z * R_{23}}{\delta X * R_{31} + \delta Y * R_{32} + \delta Z * R_{33}} \qquad (4.3\ 1)$$

where R is the rotation matrix:

$$R = \begin{bmatrix} \cos\phi\ \cos\kappa & \cos\omega\ \sin\kappa + \sin\omega\ \sin\phi\ \cos\kappa & \sin\omega\ \sin\kappa - \cos\omega\ \sin\phi\ \cos\kappa \\ -\cos\phi\ \sin\kappa & \cos\omega\ \cos\kappa\ \text{-sin\&}\ \sin\phi\ \sin\kappa & \sin\omega\ \cos\kappa + \cos\omega\ \sin\phi\ \sin\kappa \\ \sin\phi & -\sin\omega\ \cos\phi & \cos\omega\ \cos\phi \end{bmatrix} \qquad (4.32)$$

In the above mathematical model, the observations are the two sets of points (X_N, Y_N) and (Xo, Yo). The unknowns in this case are the corrections on the exterior orientation parameters (? X, ? Y, ? Z, ? ? , ? f , ? ?) . Through these corrections we attempt to "absorb" change that might be detected and express it through orientation errors. A least squares solution can be applied based on the above setup of observations, parameters and model. Several statistical estimates can be used to decide whether this systematic error exists or not, such as the statistical error estimate for each parameter and the actual parameter correction. If such errors are detected, the related metadata in the MIS which accompany the specific source, are updated to provide reasoning for future solutions.

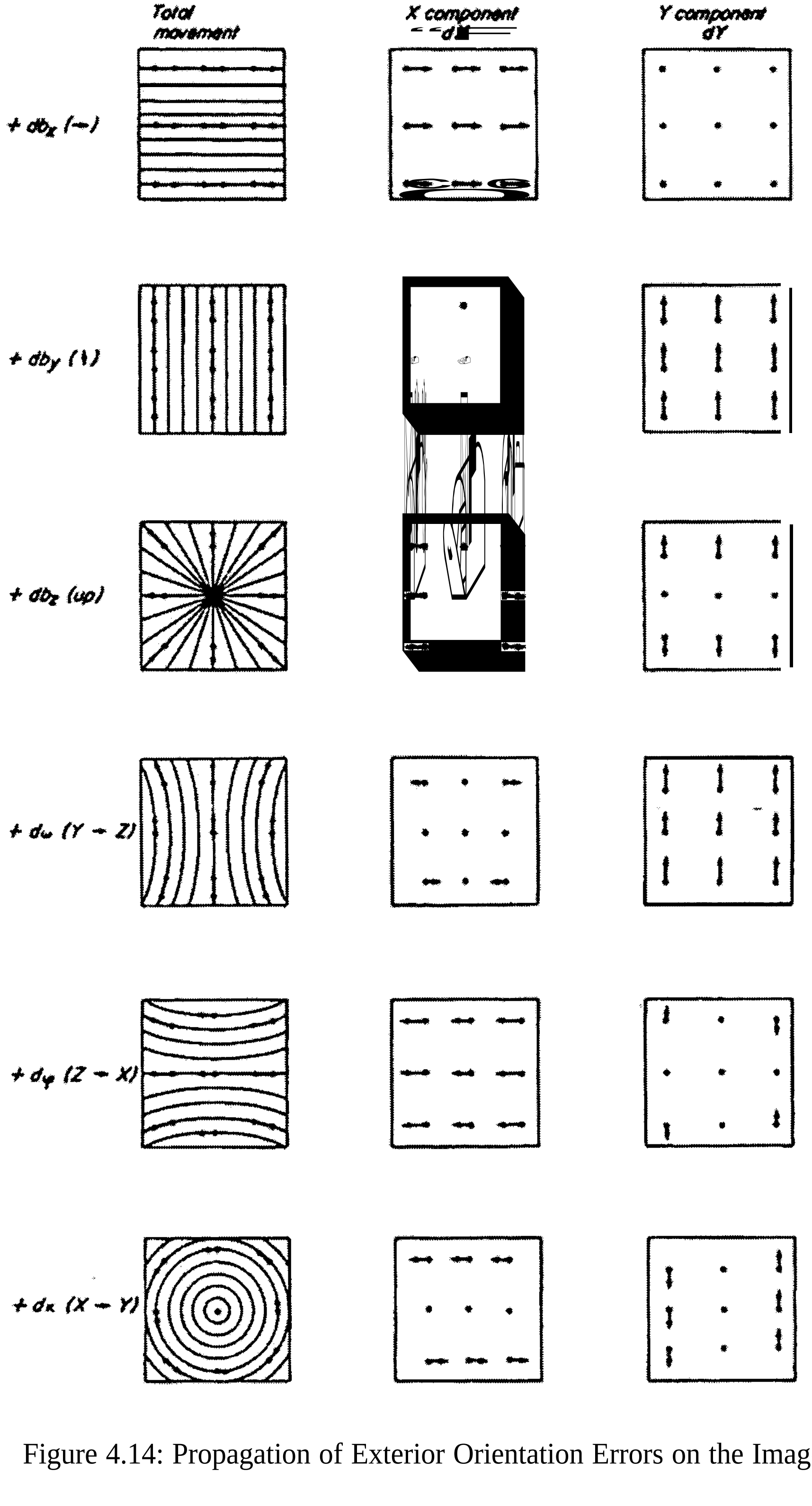


Figure 4.14: Propagation of Exterior Orientation Errors on the Image Space

4.4.2 Interior Orientation Corrections

Radial distortion is among the errors that interior orientation is dealing with. This distortion is due to the fact that it is practically impossible to make lenses with surfaces that have the perfect form of rotated parabola. Instead, the lenses'surfaces are close to spherical ones. This results in a radial, and symmetric distortion around the primary point, known as Seidel distortion (Fig. 4.15).

The formula that calculates the Seidel distortion is (Conrady, 1919):

$$? r = K_0 * r + K_1 * r^3 + K_2 * r^5 + K_3 * r^7 + \ldots \qquad (4.33)$$

The K_0 constant expresses a scaling factor that is already included in the previous solution through a shift along the Z axis. In most photogrammetric procedures, the constants K_1, K_2 are used. A higher degree polynomial (if K_3 is included) is only used where the accuracy expectations are superior. So in our case the formula used is:

$$? r = K_1 * r^3 + K_2 * r^5 \qquad (4.34)$$

In this formula r's and ? r's are the observations, where r's come from the new image coordinates (X_N , Y_N),

$$r^2 = r_N^2 = X_N^2 + Y_N^2 \qquad (4.35)$$

and ? r's are the residuals of the two radial distances,

$$? r = r_O - r_N \qquad (4.36)$$

$$\text{where } r_O^2 = X_O^2 + Y_O^2 \qquad (4.37)$$

Unknowns in this case are the constants K_1, K_2. A least squares solution, based on (Karras et al., 1998), can be applied. A decision whether this systematic error was captured through our change detection algorithm is made based on the statistical estimates that show how well this model fits to our observations. If so, like in the case of the relative orientation, the source’s metadata are updated to eliminate such errors in future use of that image.

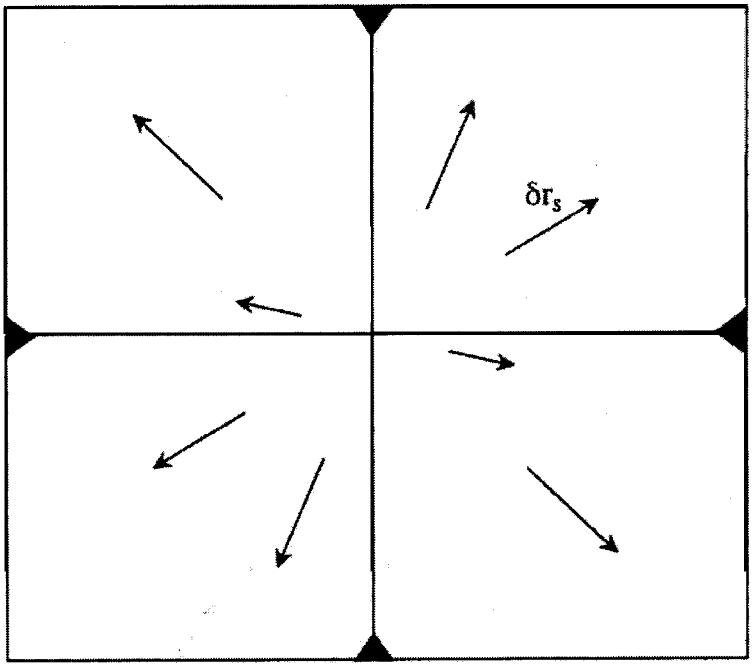


Figure 4.15: Seidel Distortion

4.5 Summary

The major goal of this thesis is to develop a digital image analysis approach that identifies automatically object outline changes in sequences of digital images. During this effort, the need to create a spatiotemporal model that would interact with the algorithm was realized. In this chapter we show how the change detection algorithm interacts with the components of our model. The change detection approach is based on least squares template matching. We extend this to function in a differential mode. In doing so, we integrate object extraction from digital imagery with change detection in a single process.

By using image orientation parameters and positional data we can reduce the problem of 3-D object monitoring to an image space 2-D matching problem. In this hybrid approach, area and feature matching are combined in one step, since raster and vector datasets are integrated to enhance our solution. Analysis of the edge representation within a template, before the actual matching takes place, extends the use of older raster information. As a post-process, actual change is distinguished from different representations of the same object due to sensor inaccuracies, through fitting models of known systematic photogrammetric errors in the exterior and interior orientation process. The obtained results provide feedback to the corresponding components of our model through the new change information.

Chapter 5

Automated Spatiotemporal Change Detection in Digital Aerial Imagery: Experiments and Evaluation

In this chapter we present the implementation method followed in order to develop a prototype for our change detection approach. Experiments are introduced and an evaluation of the performance of the algorithm is attempted.

5.1 Experiments

5.1.1 Implementation

In order to test the change detection algorithm performance a prototype spatiotemporal gazetteer was created of a semi-urban scene (campus of UMaine). Our multimedia sources provide a description of the campus during the last century. The available datasets are:

- six aerial imagery datasets representing the last half of the century (1949, 1956, 1962, 1971, 1985, and 1997)
- the corresponding vector datasets of extracted outlines on the above images.

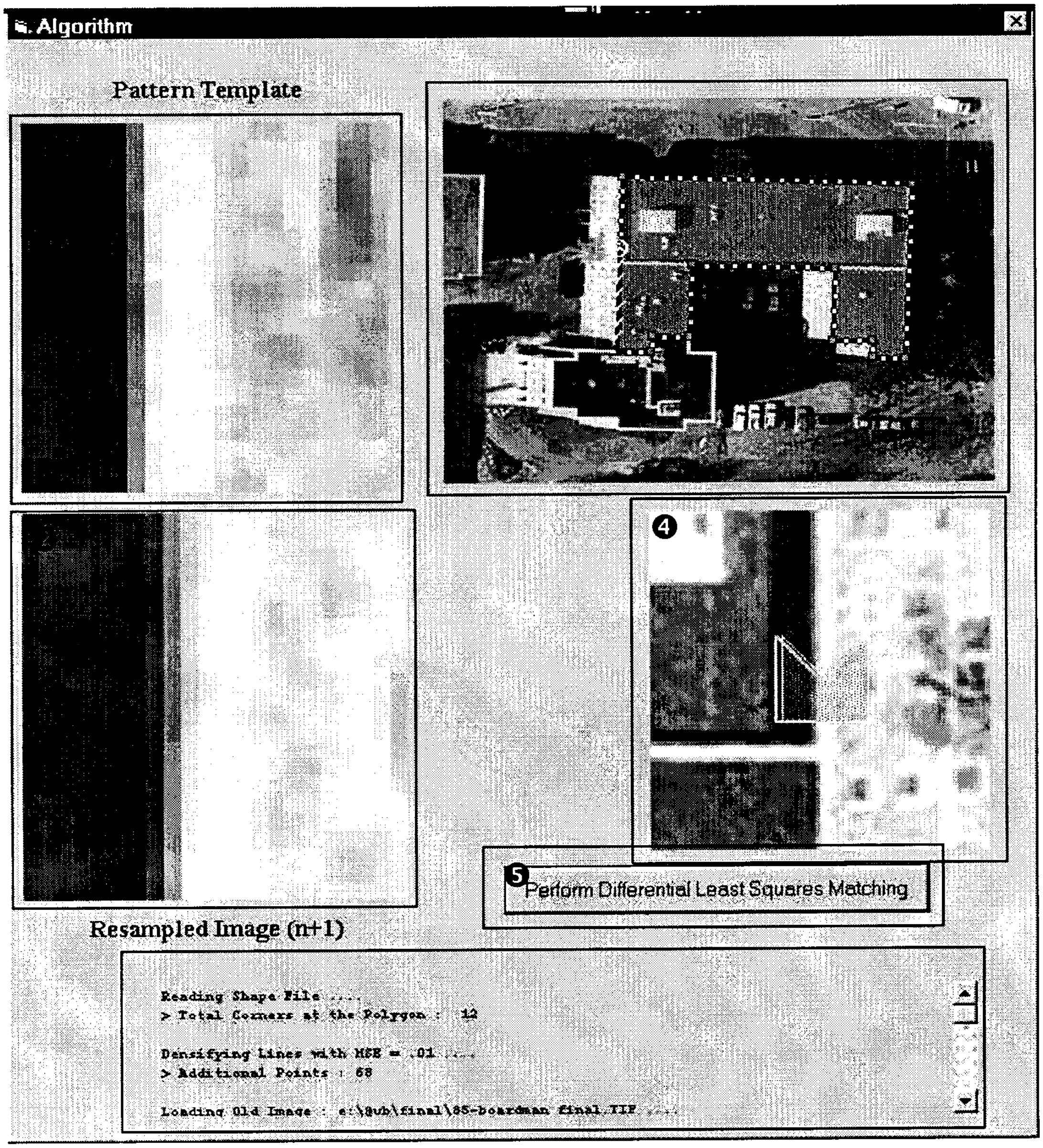
The prototype implementation was developed using Microsoft's Visual Studio on a stand-alone PC with the following configuration:

• CPU:	PII-450MHz
• RAM:	128MB
• Display Adaptor:	8 MB
• Hard Disk:	18 GB

ESRI's MapObjects ActiveX component was used to visualize the results. Input into our application were the following:

- Vector file representing extracted object in time T-dT (ESRI's Shapefile format)
- Raster file representing the object in time T-dT (TIF format)
- Raster file representing the object in time T (TIF format)
- Window size, expressing vector accuracy and object elasticity
- Minimum Spatial Element Value.

A screenshot of our application environment can be seen in figure 5.1. After the DLSM solution, each matching point was assigned a boolean value, showing if matching was successful or not. A workflow of our algorithm implementation is shown in figure 5.2.



```
Reading Shape File ....
> Total Corners at the Polygon :  12

Densifying Lines with MSE = .01 ....
> Additional Points : 68

Loading Old Image : e:\Sub\final\85-boardman final.TIF ....
```

Figure 5.1: Application Prototype

Legend:

Old Image Pattern (size n x n)
❷ New Image Pattern (size (n+l) x (n+l))
Control Panel showing DLSM Progress
❹ Trajectory of Pattern Window
❺ Command button to start the execution
Log File showing progress
Vertical Lines showing the geometry of the edge

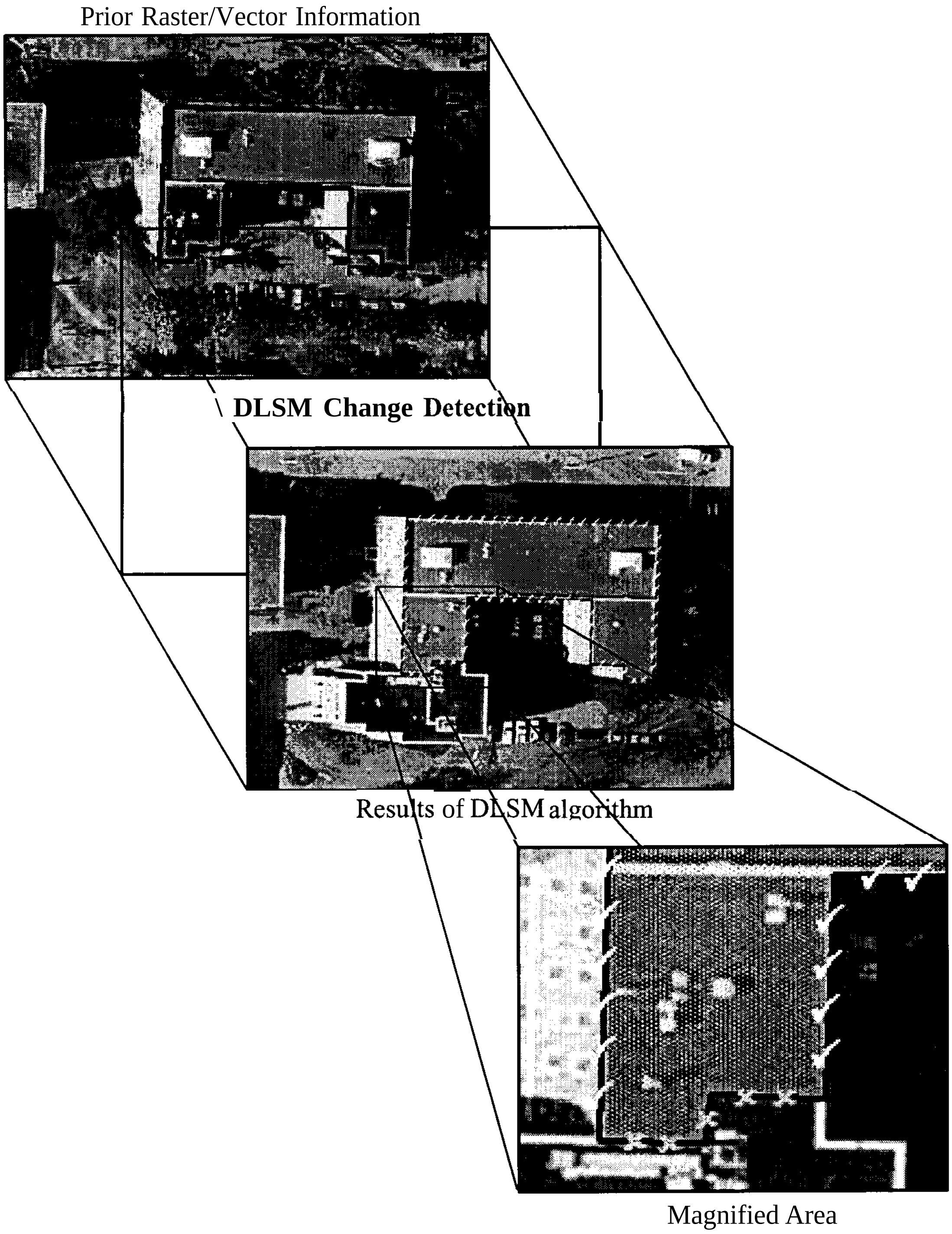


Figure 5.2: Example of our Algorithm Implementation, where:

× Unsuccessful match.

✓ Successful match.

51.2 Edge Representation

Before the differential least squares matching, a pre-process is performed on imagery from previous time instance to extract information for the edge representation in the raster domain. As previously mentioned, the average first derivative of each column is calculated along the direction perpendicular to the edge. Then a filtering method is introduced to extract peaks that might represent edges. During this step, three thresholds were considered.

i. The first one (C_1) expresses a threshold, beyond which the relative sharpness reveals a strong edge. Experiments showed that a value of 70% should be assigned.

ii. The second constant C_2 expresses a threshold, below which the relative sharpness reveals a weak edge. This value was set to 30%.

iii. The third constant (C_3) is used where the first two fail. It detects sudden peaks based on a second derivative analysis. This constant was set empirically to the value of *70%*.

In the above filtering a comparison is performed between the dominant edge and the other gradient maxima to eliminate false responses. A second filtering is introduced based on the first derivative along the edge. The concept behind this is that an edge should have high gray value variations perpendicular to itself but at the same time show low variations along itself. During this process two thresholds are introduced:

i. the *DX-threshold,* showing the difference in the gray values along the X axis that might not imply an edge, and

ii. the *DY-threshold,* expressing the difference in the gray values along the Y axis that might not imply homogeneity.

Experiments showed that a value of 40 gray values should be assigned to *DX-threshold*, and a value of 7 gray values to the *DY-threshold.*

5.1.3 Formulation of Weight Matrix

The analysis of the edge representation is incorporated in the matching process through the weight matrix. As described in the theoretical section the constant a expresses a scale-space factor in the formulation of the weights. Experiments showed that for an average resolution of *1/5000*, a value close to 0.4 should be assigned to a. This corresponds to a value of *d/3* for s_1 and *2d/5* for s_2, where d is the distance between the two means of the distributions for an edge representation similar to figure 4.7. Note that $s_2 > s_1$ to compensate for the fact that the original edge we try to match (vector edge) corresponds to s_2 so pixels around it should contribute more to the solution.

A scaling factor describing the expected geometry of the edge was introduced in Chapter 4. It is applicable in cases where curbs at the edges of buildings exist. It is expressed through a combination of two identical Gaussian Distributions. The distance between each mean and the vector edge is the actual width of the expected curb. In our case a curb of *8 inches* was assigned. In addition to this, a standard deviation value of 5.3 *inches* was assigned for each distribution. This value was chosen so that the weights are scaled down by 40% at the distance of the standard deviation (5.3 inches).

51.4 Matching Decision Support

During this execution of the matching loop, several criteria are considered to support the decision whether a matching is successful or not.

- **Shift dX**

First we check for the value of the shift dX. A threshold of *1/10* of a pixel is set to characterize a matching as successful. During our experiments with the application of the DLSM, a significant amount of visually successful matches was not correctly identified, because the value of the shift was not close to our threshold. After examining the behavior of dX during the iterations, several periodicity patterns were detected (Fig. 5.3). In many cases though these patterns were misleading, since they appear in a high amount of unsuccessful matches. The distinction between a successful match and an unsuccessful one is the range of dX. If that variation is within a pixel value, then it is considered successful, otherwise rejected.

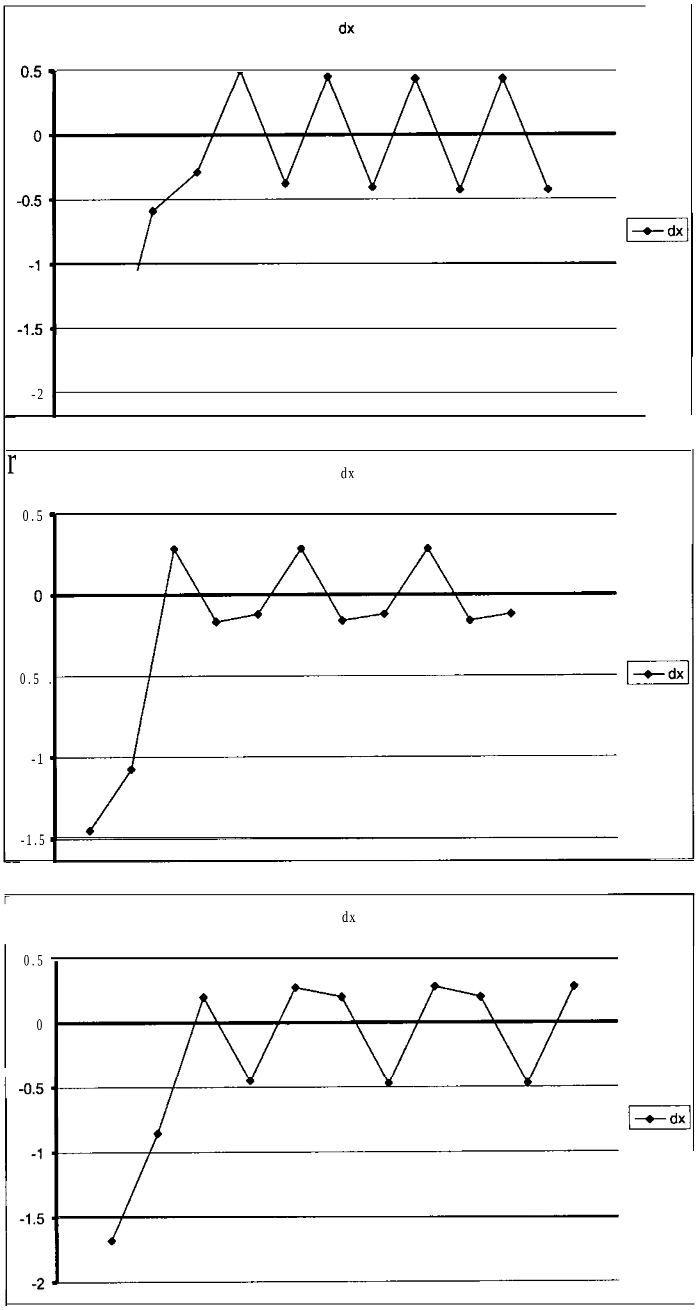


Figure 5.3: Patterns of dX Variations through Iterations

- A posteriori variance of unit weight

This variance expresses how well the overall model fits to the observations. When a value less than 0.5 was obtained, a successful match is indicated.

- Variance-covariance matrix for parameters

This variance shows a statistical estimation for the accuracy of the obtained results. An empirical value of 20% of the a posterior-i variance of unit weight was assigned.

5.2 Evaluation

5.2.1 Cost of Application

Before we discuss the advantages of this approach, a description of the application cost should be given. Here we should note that the goal of this thesis was to create a prototype that would provide the ability to test the underlying image analysis theory, rather than introducing a fully functional, optimized environment. Keeping that in mind as well as that the available CPU power when this thesis was submitted was twice the CPU power used, a time cost evaluation was performed. When all the available computer resources were used, the average time per matching point was approximately 5 sec. It should be mentioned that when the number of iterations was reaching the number of fifteen, then the process stopped and the result was classified as an unsuccessful one. So no point could take more than 8 sec. Experiments were also performed with and without the analysis of the edge representation. This geometric analysis combined with the incorporation of expected geometry proved to be very fast adding no significant time (estimated < 0.05 sec). This results from the fact that the weight matrix has to be constructed once for every matching point.

If this analysis is expanded to an object level, then it becomes evident that the average time per feature is directly related to the resolution of change, as introduced in the query process. The Minimum Spatial Element defines the number of points used per line. For a value of 3 meters, the average number of points expressing the average building size was around 70. So based on a 5 sec average time per point, an average feature matching is completed in 350 sec, almost 6 min. This number raises questions for the application of the algorithm in real-time mode. A more appropriate usage would be in an environment that is continuously updated, but does not require high speed of response. Nevertheless, these time constraints are overshadowed by the reliability and accuracy as described in the following section.

5.2.2 Benefit of Application

Our main objective was to correctly identify changed areas, and to verify unchanged points. Three factors were considered through this evaluation, namely good localization, clear response and good detection. These factors are broadly accepted in the object extraction community and characterize a successful result.

5.2.2.1 Good Localization: minimal distance between the detected edge and real edge.

Good localization is the major advantage of LSM, as proven from previous applications in the photogrammetric field. Indeed, experiments showed that the extracted edges approximated the real ones at subpixel accuracy of 0.1 *pixel.* Such accuracy is difficult to be obtained and establishes high reliability in the performance of the algorithm.

5.2.2.2 Clear Response: only one response per edge.

During the experimental process a large number of edges were not correctly identified. After investigating the radiometric representations of those points, a common error factor was found. There were some neighbor edges, parallel to the desired one, that were guiding the matching process in a wrong way. To compensate for that, a prior analysis of the edge representation was introduced.

A significant improvement in performance was noticed when the edge levels were three or more and we were able to establish a geometric representation of the edge. The width of the level(s) establishes a scaling factor for the whole process, which guides the template quickly and accurately to the new edge when there is no change. With this, we achieve one response from multiple edges. Higher accuracy was also achieved when edge geometry was incorporated in the solution through the statistical analysis of the results (a posteriori variance of unit weight, variance-covariance matrix for parameters).

5.2.2.3 Good Detection: the ability to locate and mark all the real edges.

A large number of errors was corrected by the identification of levels. Even though that provided a boost in the performance and reliability of our technique, in some specific cases, this analysis was inserting non-desired edges in the process. Random noise such as building windows, shadows, or cars was initially affecting the algorithm. The *expected* edge geometry analysis allows us to distinguish random noise from real edge.

Based on this, we scale down pixel weights that are not expected to be part of the real representation of an edge, and we increase the weights of the ones expressing real edges. Especially in the case of shadows, a problem inherently difficult since the artificial edges are very strong, the expected geometry of the edge, expressed through the size of the curb, introduced the essential metric reasoning to compensate for such errors.

5.3 Summary

In this chapter implementation issues of our proposed change detection approach were discussed. Experiments showed that this implementation might be costly in terms of execution time, but the obtained results surpass this drawback. The results were evaluated for good localization, clear response and good detection.

Good localization was achieved through an accuracy of 0.1 pixel. This was expected since it is the major advantage of least squares matching. For the other two factors, a significant improvement was noticed when the edge levels were three or more and we were able to establish a geometric representation of the edge. This geometric analysis proved to be very fast, since the weight matrix is constructed only once for every matching point. The width of the level(s) establishes a scaling factor in the whole process, which guides the template quickly and accurately to the new edge, when there is no change. Edge geometry analysis allows us to distinguish random noise from the real edge through the manipulation of the weight matrix. It is expressed through the size of the curb, and introduces the essential metric reasoning to compensate for errors such as shadows. This is a major advantage since this problem is inherently difficult since the artificial edges are very strong.

Chapter 6

Conclusions and Future Work

In this thesis, an approach for detecting and handling change within a broad spatiotemporal environment has been presented. Issues that address the structure, components, and information flow of this environment have been discussed. A multi-resolutional design approach has been introduced that allows the users to define their own resolutional working space. A change detection image analysis technique that interacts with the model and a querying method has been developed.

In the next sections a summary of the thesis is provided, followed by the major findings and results. Future work showing the way for improvements and expansion of the presented effort is discussed.

6.1 Summary of Thesis

This thesis addresses the issue of image-based change detection. During this research it was realized that this process could not be isolated, but should rather be incorporated within an integrated spatiotemporal environment (Fig. 6.1). In addition to that, a change resolution model should be developed that will distinguish meaningful changes based on user requirements.

A model for change is proposed that establishes a general framework for the incorporation of image analysis techniques. This model is based on the SpatioTemporal

Gazetteer (Beard and Agouris, 1997) as a model that makes more effective use of multiple information resources. Theoretical analysis is introduced focusing on the choice of reference system for spatiotemporal information, and on change dimensionality and representation issues within a spatiotemporal environment. A change-oriented model is proposed. Implementation of this analysis generated the *Differential Spatiotemporul Gazetteer.* This modified version of the original gazetteer is composed of a Geographic Identity Information Level, a Change Indexing Information Level and a collection of child sources at the Child Information Level. Mathematical dependency between these components is presented.

We modify the original model to incorporate the essential tools that extract change and model behavior. In doing so, an expansion to a *broader spatiotemporal environment* is proposed. Two new components are introduced, the Change Detection Tools and the Knowledge Base. Information flow within the environment is described and component interaction is presented, focusing on the change detection aspects.

In spatiotemporal applications, meaningful changes vary according to object type, level of detail, and nature of application (Mountrakis et al., 2000). To accommodate this, we identify in our model three levels of change, ranging from simple verification of the existence of an object to the identification of change and the detailed description of this change. In addition, the user has the flexibility to navigate through different resolution representations by manipulating two functions, the Minimum Spatial Element and the Minimum Temporal Element.

With the proposed spatiotemporal model and query method, we optimize the change detection process, by providing the correct datasets, accuracy during the detection process, and a framework for storing the obtained results. The digital image analysis method that was developed automatically identifies object outline changes in sequences of digital images. This change detection technique is based on least squares template matching (LSM). We extend LSM to function in a differential mode. In doing so, we integrate object extraction from digital imagery with change detection in a single process. By using image orientation parameters and positional data we can reduce the problem of 3-D object monitoring to an image-space 2-D matching problem.

In this hybrid approach, area- and feature-based matching are combined in one step, since raster and vector datasets are integrated to enhance our solution. Analysis of the edge geometry within a template, before the actual matching takes place, improves the accuracy and reliability of the presented technique. As a post-process, object change is distinguished from different representations of the same object due to image and sensor inaccuracies, through fitting models of known systematic photogrammetric errors in the exterior and interior orientation process, respectively. The obtained results provide feedback to the corresponding components of our spatiotemporal model through the detected change information.

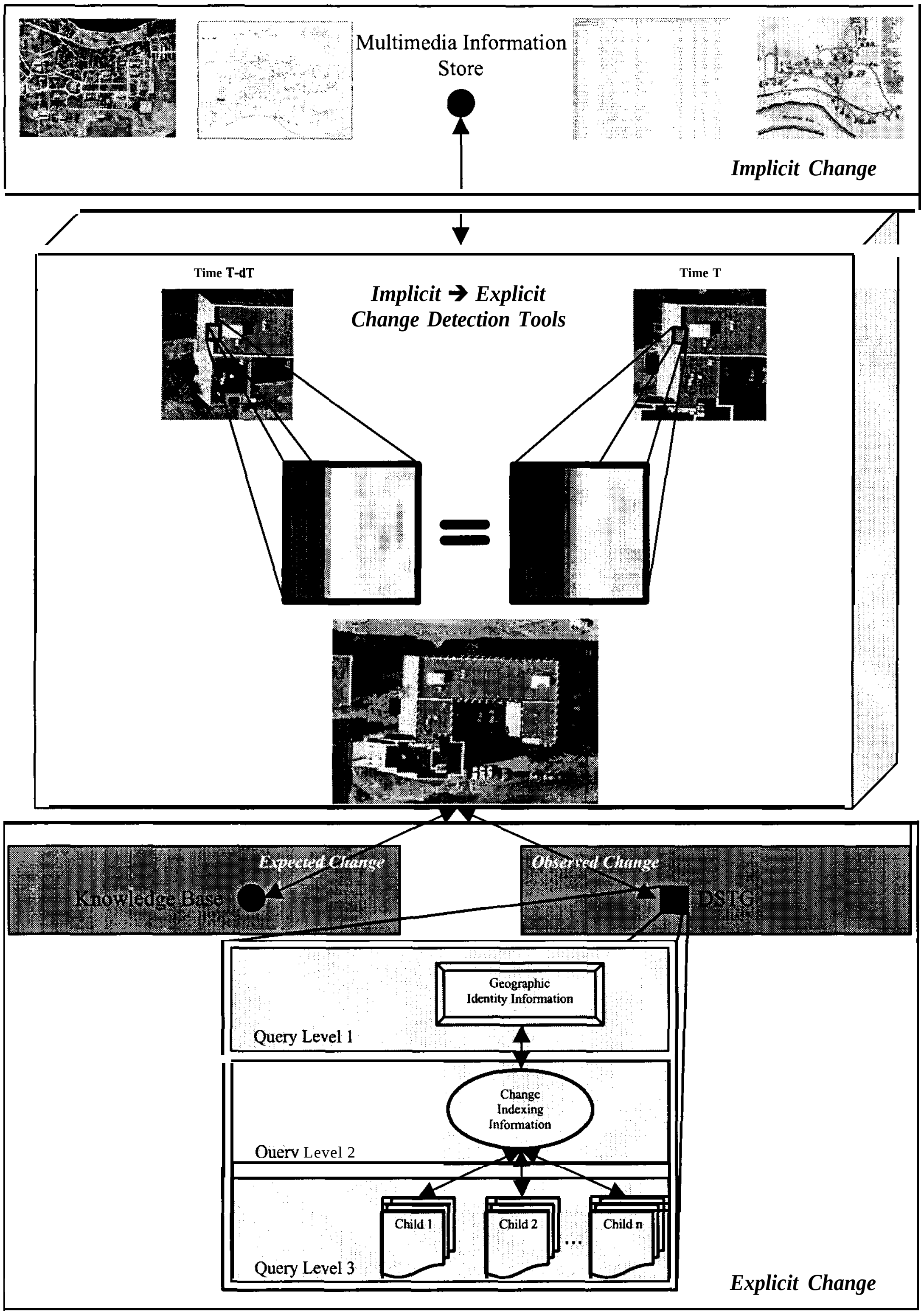


Figure 6.1: Proposed Spatiotemporal Environment

6.2 Results and Major Findings

The major results of this thesis are:

I. *Establishment of a spatiotemporal environment that allows the integration of multi-temporal, multi-resolutional, multi-type information for image-based change detection.*

In our proposed environment change is saved explicitly and not implicitly through different object representations. More specifically, we store only the dimensions of change that are modified. With this approach we can ensure minimal redundancy by reducing a multi-dimensional problem to its minimal modified dimensions and maximum efficiency because change is directly available within our environment. Explicit change information is handled by the Differential Spatiotemporal Gazetteer.

To incorporate change detection and behavior modeling, the gazetteer is expanded to a broader spatiotemporal environment. Two new components are introduced, the Change Detection Tools and the Knowledge Base. Information flow within the environment justifies this structure, since multiple types of information are available for the image analysis technique.

II. *User-defined resolution of spatiotemporal change.*

The communication of spatiotemporal change requires the consideration of users who have diverse needs, and exhibit various levels of understanding of the content and of the databases that they access. To accommodate this, we develop a method that provides users with the ability to define their own resolution of interest. Three levels of change are recognized: Existence of Object, Existence of Object Change and Characteristics of Object Change. These levels correspond to the three Information Levels of the conceptual design of the model, namely the Geographic Identity Information Level, the Change Indexing Information Level and the Child Information Level.

III. Detect image-based changes in building outlines by combining prior knowledge and object extraction in a single process.

With the proposed spatiotemporal model, we improve the change detection process, by providing precision in the datasets used, accuracy during the detection process, and a framework for storing the obtained results. The digital image analysis method that was developed automatically identifies object outline changes in sequences of digital images. The change detection technique is based on least squares template matching. We extend this to function in a differential mode and incorporate prior information. In doing so, we integrate in a single process object extraction from digital imagery with change detection.

During the matching procedure, the statistical origin of least squares contributes to a robust evaluation of the performance. Analysis of the edge geometry within a template before the actual matching takes place improves the accuracy and reliability of the presented technique. By applying reasoning at the output level, actual changes are distinguished from misregistration errors through a directional pattern analysis. If such errors are detected, the update of the corresponding metadata of the source contributes to a more reliable future use of that image. The obtained results provide feedback to the corresponding components of our spatiotemporal model through the new change information.

6.3 Future Work

Starting from the spatiotemporal model there are some theoretical issues that could be addressed. The expansion of our multi-resolutional change decomposition to include part-whole relations would be a challenging task. It would allow more complex modeling of reality and could improve the communication of the environment with the user. Based on that, a different setup for child databases can be introduced. Also higher level reasoning could be achieved through the manipulation of complex objects and interactions/relations between them. Relations such as association and aggregation would be an example.

Another indexing structure can be created over the child databases to improve performance in the change detection process. This index would store object representations based on change accumulations. With this object representations could be calculated faster in the input process of the change detection.

The incorporation of conceptual entities in the modeling would increase the applicability of this approach. Spatiotemporal objects under examination would not be restricted to physical entities. This would permit a more ‘human-like” methodology, where cognitive reasoning can be applied.

The development of the Knowledge Base component can add significant reasoning to the process. It can act as a validity check component for the obtained results and facilitate future change detections. The spatiotemporal resolution hierarchies can be handled in this component. Different hierarchies can be merged, combined or expanded. This would require the development of an ontology-based approach.

In the image processing domain the proposed approach can be extended in several ways. First the input dataset can be expanded to include other types of imagery. Our experiments focused on vertical photography, but the mathematical foundation of our algorithms would accommodate equally well oblique imagery, even close range photos. The only constraint in this process would be that the image datasets used project the 3-D reality from the same viewpoint into the 2-D image spaces. Satellite imagery can also be incorporated by matching multi-layer windows (cubes).

Within the matching process each point is matched locally and independently. Part of the future work would be to proceed towards an object-wise global solution where solutions along numerous points of the same outline are connected. This would strengthen the solution, by identifying random variations that may appear in the new scene. These variations can be either local (e.g. shadows), or global (e.g. illumination differences or due to different sensor radiometric resolution). Moreover, the edge geometry analysis technique can be extended to include semantic reasoning based on scale space concepts. This will enable us to process multi-scale imagery more efficiently.

This approach can be also expanded to provide the ability to detect changes from other types of sources, such as scanned maps and vectorized data (CAD). This would allow a universal multi-source approach to detect change and would take full advantage of the existing diverse information of the spatiotemporal environment.

Future development of the image analysis technique could address change detection in a more detailed level. Currently an outline is compared on two images without considering changes within or outside that. A region growing approach can be incorporated to analyze the area inside the outline, and detect possible variations that do not affect the outline itself. This region growing technique would also empower the reliability of the currently obtained results by verifying them through an “internal energy” function.

References

Ackerman F. (1984) Digital Image Correlation: Performance and Potential Application in Photogrammetry. *Photogrammetric Record*, 11(64), pp. 429-439.

Adelson E. H. and J. R. Bergan (1985) Spatiotemporal Energy Models for the Perception of Motion. *Journal of the Optical Society of America*, A, vol. 2, no. 2, pp. 284-299.

Aggarwal J.K. and N. Nandhakumar (1988) On the computation of motion from sequences of images - A review. *Proceedings of the IEEE*, 76(8), pp. 917-935.

Agouris P. (1992) Multiple Image Multipoint Matching for Automatic Aerotriangulation. The Ohio State University, Columbus, Ohio.

Agouris P. and A. Stefanidis (1998) Intelligent Image Retrieval from Large Databases using Shape and Topology. *IEEE-ICIP98 (International Conference on Image Processing)*, Chicago, Ill, Vol. 2, pp. II/779-II/783.

Agouris P., G. Mountrakis, and A.Stefanidis (2000b) Automated spatio-temporal change detection in digital aerial imagery. *Proceedings of Aerosense* 2000, To appear.

Agouris P., G. Mountrakis, K. Beard, and A. Stefanidis (2000a) Modeling and Detecting Change in an Integrated Spatio-Temporal Environment. *International Archives of Photogrammetry and Remote Sensing,* Amsterdam 2000, To appear.

Agouris P., K. Beard, G. Mountrakis, and A. Stefanidis (2000c) Capturing and Modeling Geographic Object Change: A SpatioTemporal Gazetteer Framework. *Photogrammetric Engineering and Remote Sensing,* To appear.

Albano G., K. Gheli, G. Occhiuto and R. Orsini (1985) Galileo: a strongly typed interactive conceptual language. ACM *TODS*, Vol 10, No. 2.

Allen E., G. Edwards, and Y. Bedard (1995) Qualitative Casual Modeling in Temporal GIS. *Proceedings of the International Conference on Spatial Information Theory (COSIT95)* , Lecture Notes in Computer Science Vol. 988, Berlin, Springer-Verlag.

Allen J. (1983) Maintaining knowledge about temporal intervals. *Communications of the ACM* 26(11), pp. 832-843.

Al-Taha K. and R. Barrera (1990) Temporal data and *GIS:* an overview. In: *GIS/LIS 90* , Anaheim, CA, pp. 244-254.

Anderson T. L. (1982) Modeling Time at the Conceptual Level. In P. Scheuermann, editor, *Proceedings of the International Conference on Databases: Improving Usability and Responsiveness,* Jerusalem, Israel, Academic Press, pp. 273-297.

Ariav G. (1986) A temporally oriented data model. *ACM Transactions on Database Systems,* vol. 11, no. 4, pp. 499-527.

Armstrong M. (1988) Temporality in spatial databases. In: *GIS/LIS* 88 , San Antonio, TX, pp. 880-889.

Baltsavias E.P. (1991) Multiphoto Geometrically Constrained Matching. *49, ETH - Institute for Geodesy and Photogrammetry,* Zurich.

Barbic F. and B. Pernici (1985) Time Modeling in Office Information Systems. In S. Navathe, editor, *Proceedings of ACM SIGMOD International Conference on Management of Data,* Austin, TX, Association for Computing Machinery, pp. 51-*62.*

Barnard S.T. and W.B. Thompson (1980) Disparity analysis of images. *IEEE-PAMI*, *(2) 4,* pp. 333-340.

Batty M. and Y. Xie, (1994a) Modeling Inside GIS:Part 1. Model Structures, Exploratory Spatial Data Analysis and Aggregation. *International Journal of GIS*, *8, 3.*

Batty M. and Y. Xie, (1994b) Modeling Inside GIS:Part 2. Selecting and Calibrating Urban Models Using Arc/Info. *International Journal of GIS*, 8, 3.

Beard K. and P. Agouris (1997) A Spatio-Temporal Model for Integrated Information Management, Project Proposal for NIMA (National Imagery and Mapping Agency), http://www.spatial.maine.edu/~peggy/NIMA.html.

Beard M. K and T.R. Smith (1997) A Framework for Meta-Information in Digital Libraries. A. Sheth and W. Klaus (Eds.), *Managing Multimedia Data: Using Metadata to Integrate and Apply Digital Data,* McGraw Hill.

Beard M. K., T. R. Smith, and L. Hill (1997) Meta-Information Models for Georeferenced Digital Library Collections. *Proceedings of the Second IEEE conference on Metadata,* Silver Springs, MD.

Becker L. and R. Guting (1992) Ruled-Based Optimization and Query Processing in an Extensible Geometric Database System. ACM Transactions on Database Systems 17(2), pp. 247-303.

de Berg M., M. van Kreveld, M. Over-mars and 0. Schwarzkopf (1997) Computational Geometry: algorithms and applications. Springer-Verlag, Berlin.

Bignone F., 0. Henricsson, P. Fua and M. Sticker (1996) Automatic Extraction of Generic House Roofs from High Resolution Aerial Images. *In Computer Vision – Proceedings* of *ECCV* 96 , Lecture Notes in Computer Science No. 1064, Springer-Verlag, Berlin, pp. 85-96.

Black M. J., S. X. Ju, and Y. Yacoob (1996) Cardboard people: A parameterized model of articulated motion. *2nd International Conference on Automatic Face- and Gesture-Recognition,* Killington, Vermont, pp. 38-44.

Burt P.J. (1984) The Pyramid as a Structure for Efficient Computation. *Multiresolution Image Processing and Analysis,* A. Rosenfeld (ed.), Springer-Verlag, New York, NY, pp. 6-35.

Buttenfield B. and R. McMaster (eds.) (1991) *Map Generalization: Making Rules for Knowledge Representation.* New York, Longmont Scientific and Technical.

de Cambray B. (1993) Three-Dimensional modeling in a geographical database. *Auto Carto* 11, Minneapolis, MN, pp. 338 -347.

Carey M., D. DeWitt and S. Vandenberg (1988) A Data Model and Query Language for EXODUS. *Proceedings of the 1988 ACM SIGMOD Conference,* Chicago.

Carswell J. (1988) A Combined Digital Image Correlation Procedure for Establishing Relative Orientation, The Ohio State University, Columbus, Ohio.

Chandra R., A. Segev, and M. Stonebraker (1994) Implementing calendars and temporal rules in next generation databases. *Proceedings of ICDE,* pp. 264-273.

Chang N.S. and K.S. Fu (1980) Query-by-pictorial-example. *IEEE Transactions on Software Engineering,* vol. SE-6, no. 6, pp. 5 19-524.

Chellappa R., Q. Zheng, L.S. Davis, D. DeMenthon, and A. Rosenfeld (1993) Site-Model-Based Change Detection and Image Registration. *DARPA93*, pp. 205-216.

Cho W. (1995) Relational matching for automatic orientation. Ph.D. thesis, Department of Geodetic Science and Surveying, The Ohio State University, Columbus, OH.

Chorochronos (2000) http://www.dbnet.ece.ntua.gr/cgi-users/choros/search.pl?showall=Show+All+Records

Chrirsman N. (1975). Topological information systems for geographic representation. *Proceedings of Second International Symposium on Computer-Assisted Cartography (Auto-Carto 2)*, Falls Church: ASPRS/ACSM, pp. 346-35 1.

Chrisman N. (1978) Concepts of space as a guide to cartographic data structures. In Dutton, G. (Ed.) *Proceedings of the First International Advanced Study Symposium on Topological Data Structures for Geographic Information Systems,* Cambridge, MA: Harvard Laboratory for Computer Graphics and Spatial Analysis, pp. 1-19.

Ciapessoni E., E. Corsetti, A. Montanari, and P. San Pietro (1993) Embedding time granularity in a logical specification language for synchronous real time systems, *Science of Computer Programming,* 20 (1993) pp. 141-171.

Claramunt C., C. Parent, and M. Theriault (1997) Design Patterns for Spatiotemporal Processes. *IFIP* 1997, Chapman and Hall.

Clifford J. and A. Croker (1987) The historical relational data model (HRDM) and algebra based on lifespans. *Proceedings of the 3rd IEEE International Conference on Data Engineering,* pp. 528-537.

Clifford J. and A. Rao (1987) A Simple, General Structure for Temporal Domains. *Proceedings of the Conference on Temporal Aspects in Information Systems*, France, AFCET, pp. 23-30.

Codd E. (1970) A Relational Model of Data for Large Shared Data Banks. *Comm. ACM* 13:6.

Conrady A. (1919) Decentered lens system, *Monthly Notices of the Royal Astronomical Society.* Vol. 69, pp. 384-390.

Couclelis H. (1996) Towards an Operational Typology of Geographic Entities with Ill-defined Boundaries. P. Burrough and A. U. Frank (Eds.), *Geographic Objects with Indeterminate Boundaries,* Taylor and Francis, pp. 45-55.

Cross M.C. and Y. Tu (1994) Defect Dynamics for Spiral Chaos in Rayleigh-Benard Convection. *Technical report*, California Institute of Technology.

Del Bimbo, A. E. Vicario, and D. Zingoni (1995) Symbolic Description and Visual Querying of Image Sequences Using Spatio-temporal Logic. *IEEE Transactions on Knowledge and Data Engineering,* Vol. 7, No. 4, pp. 609-621.

Doom B., P. Agouris, R. Al-Tahir, A. Stefanidis, and 0. Zilberstein (1990) Digital Stereo Matching. *In Perspective.* 10, The Ohio State University, Columbus, Ohio.

Dumas M., R. Lozano, M.C. Fauvet, H. Martin, and P.C. Scholl (2000) Orthogonally modeling video structuration and annotation, *Proceedings of the AAAI Workshop on Spatial and Temporal Granularities,* Austin TX (USA).

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

Egenhofer M. and R. Golledge (1998) *Spatial and Temporal Reasoning in Geographic Information Systems.* Oxford University Press, New York.

Ehleringer J. and C. Field (eds.) (1993) *Scaling Physiological Processes, Leaf to Globe.* New York, Academic Press, Inc.

Embey D.W. and G. Nagy (1986) Toward a high-level integrates image database system. *Technical Report,* Image Processing Laboratory, Rensselaer Plolytechnic Institute, Troy, NY.

Erwig M. and M. Schneider (1999) Developments in Spatio-Temporal Query Languages, *IEEE International Workshop on Spatio-Temporal Data Models and Languages,* pp. 441-449.

Erwig M., M. Schneider, R. H. Güting, and M. Vazirgiannis (1999) Spatio-Temporal Data Types: An Approach to Modeling and Querying Moving Objects in Databases. *GeoInformatica*, 3(3), pp. 265-291.

Esperanca C. and H. Samet (1997) An overview of the SAND spatial database system. *Communications of ACM.*

Euzenat J. (1995) An algebraic approach for granularity in qualitative space and time representation. *Proceedings of IJCAI*, San Mateo, CA, pp. 894-900.

Fisher A., T. Kolbe and F. Lang (1999) On the use of geometric and semantic models for component-based building reconstruction. In *SMATI 1999proceedings.*

De Floriani L., P. Magillo, E. Puppo, M. Bertolotto (1996) Variable Resolution Operators on a Multiresolution Terrain Model. *Proceedings 4th ACM Workshop on Advances in Geographic Information Systems (ACM-GIS '96)*, RockVille, Maryland, November 1996, pp.121-128.

Forstner W. (1982) On the geometric precision of digital correlation, International Archives of Photogrammetry & Remote Sensing (24) 3, pp. 176-l 89.

Förstner W. (1986) A feature based correspondence algorithm for image matching, *International Archives of Photogrammetry & Remote Sensing, (26)* 3/3, pp. 150-166.

Frank A. (1998) Different Types of "Times" in GIS . *In Spatial and Temporal Reasoning in Geographic Information Systems.* M. Egenhofer and R. Golledge (eds), Oxford University Press, New York, 1998, pp. 40-62.

Frank A. U., and D. M. Mark (1991) Languages issues for GIS. *In Geographic Information Systems: Principles and Applications.* Maguire, D. J., Goodchild, M. F., and Rhind, D. W. (eds.), Vol. 1, pp. 147-163.

Freksa C. (1992) Temporal reasoning based on semi-intervals. *Artzjkial Intelligence,* 54: 199-227.

Galton A. (1993) Towards an Integrated Logic of Space, Time, and Motion. *Proceedings of the 13th International Joint Conference on Artzjkial Intelligence (IJCAI'93)*, Chambery, France, August 1993, pp. 1550-1555.

Goodchild M. (1998) Towards a GeoLibrary. *Computers and Geosciences.*

Goodchild M. and J. Proctor (1997) Scale in a digital geographic world. *Geographical and Environmental Modelling* l(1): 5-23. [275].

Greenfeld J.S. (1987) A Stereo Vision Approach to Automatic Stereo Matching in Photogrammetry. Ph.D. Dissertation Thesis, The Ohio State University, Columbus, Ohio.

Greenfeld J.S. and A.T. Schenk (1989) Experiments with Edge-Based Stereo Matching. *Photogrammetric Engineering and Remote Sensing,* 55(12), pp. 1771-1777.

Grimson W.E. (1985) Computational Experiments with a Feature-Based Stereo Algorithm. *IEEE PAMI*, 7(1), pp. 17-34.

Gruen A. (1985) Adaptive Least Squares Correlation: A Powerful Image Matching Technique. *South African Journal of Photogrammetry,* Remote Sensing and Cartography, Vol. 14, No. 3, pp. 175-187.

Gruen A. and D. Stallmann (1993) High Accuracy Edge Matching with an Extension of the MPGC-Matching Algorithm. *Applications of Geodesy to Engineering,* AG Symposia Vol. 108, Springer-Verlag, pp. 339-350.

Gruen A. and P. Agouris (1994) Linear Feature Extraction by Least Squares Template Matching Constrained by Internal Shape Forces. *International Archives of Photogrammetry & Remote Sensing,* Vol. XXX, Part 3/1, pp. 316-323.

Gruen A., E. Baltsavias and 0. Henricsson (eds.), (1997) *Automatic Extraction of Man-Made Objects from Aerial and Space Images II,* Birkhaeuser Verlag, Basel.

Gruen A., H.G. Maas and A. Stefanidis (1994) Feature Tracking in 3-D Fluid Tomography Sequences. *IEEE-ICIP94 (International Conference on Image Processing)* , Austin, TX, Vol. 1, pp. 530-534.

Gruen A., 0. Kuebler and P. Agouris (eds.), (1995) *Automatic Extraction of Man-Made Objects from Aerial and Space Images.* Birkhaeuser Verlag, Basel.

Gruen A.W., and E.P. Baltsavias (1987) Geometrically Constrained Multiphoto Matching, *Photogrammetric Engineering and Remote Sensing,* 54(5), pp. 633-641.

Haala N. and C. Brenner (1999) Virtual City Models from Laser Altimeter and 2D Map Data, *Photogrammetric Engineering and Remote Sensing,* Vol. 65, No. 7, pp. 785- 795.

Hannah M.J. (1989) A system for digital stereo image matching. *Photogrammetric Engineering and Remote Sensing, (55)* 12, pp. 1765-1770.

Hazelton N. (1991) Extending GIS to include dynamic modeling, In *Proceedings of the Third Colloquium of the Spatial Information Research Center,* University of Otago, Dunedin, New Zealand, pp. 73-82.

Hazelton N. W. (1998) Some Operational Requirements for a Multi-Temporal 4-D GIS. *In Spatial and Temporal Reasoning in Geographic Information Systems.* M. Egenhofer and R. Golledge (eds), Oxford University Press, New York, 1998, pp. 63-73.

Hendrickx M., J. Vandekerckhove, D. Frere, T. Moons, and L. Van Gool (1997) On the 3D Reconstruction of House Roofs from Aerial Images of Urban Areas. In *International Archives of Photogrammetry and Remote Sensing,* Vol. 32, Part 3-4w2, pp. 87-96.

Hornsby K. (1999) Identity-based reasoning about spatio-temporal change. Ph.D. Thesis, University of Maine, Orono, ME.

Hornsby K. and M. Egenhofer (1999) Shifts in Detail through Temporal Zooming. *Tenth International Workshop on Database and Expert Systems Applications: Spatio-Temporal Data Models and Languages,* Florence, Italy, A. M. Tjoa, A. Cammelli, and R. Wagner (eds.), IEEE Computer Society, pp. 487-491.

Hornsby K. and M. Egenhofer (2000) Identity-based change: a foundation for spatio-temporal knowledge representation, In *International Journal of Geographical Information Science.* 14(3).

Huang C., J. Mundy and C. Rothwell (1994) Model Supported Exploitation: Quick look, Detection and Counting, and Change Detection. *Proceedings of the Second IEEE Workshop on Applications of Computer Vision,* Sarasota, FL, pp. 144-15 1.

Huang C., J. Mundy, and C. Rothwell (1994) Model Supported Exploitation: Quick look, Detection and Counting, and Change Detection. *Proceedings of the Second IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, pp. 144-15 1.

Hudson J. (1992) Scale in space and time. In R. F. Abler, M. G. Markus, and J. M. Olson (eds.), *Geography's Inner Worlds: Pervasive Themes in Contemporary American Geography.* New Brunswick, NJ, Rutgers University Press, pp. 280-300.

Huertas A. and R. Nevatia (1996) Detecting Changes in Aerial Views of Man-Made Structures, *ARPA96,* pp. 381-388.

Irani M., and P. Anandan (1997) A Unified Approach to Moving Object Detection in 2D and 3D Scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 20(6):577.

Karras G., G. Mountrakis, P. Patias and E. Petsa (1998) Modeling distortion of super-wide-angle lenses for architectural and archaeological applications, In *International Archives of Photogrammetry and Remote Sensing,* Vol. 32, Part 5, pp. 570-573.

Kumar A., V.J. Tsotras, and C. Faloutsos (1998) Designing Access Methods for Bitemporal Databases. *In IEEE Transactions on Knowledge and Data Engineering,* Vol. 10, No. 1, January/February 1998.

Langran G. (1993) *Time in Geographic Information Systems.* Taylor and Francis, London.

Langran G. and N. R. Chrisman (1988) A framework for temporal geographic information. *Cartographica,* 25(3): l-14.

Leban B., D. Mcdonald, and D. Foster (1986) A representation for collections of temporal intervals, *Proceedings of AAAI*, pp. 367-371.

Lécluse C. and P. Richard (1989) The 0 Database Programming Languages. *Proceedings of the 15th VLDB Conference,* Amsterdam.

Lillestrand R. (1972) Techniques for Change Detection, *IEEE Transactions on Computers,* No.7, pp. 654-659.

Lindeberg (1994) *Scale-Space Theory in Computer Vision.* Kluwer Academic Publishers, Dordrecht, Netherlands.

Lombardo D. and Z. Kemp (1997) Toward a Model for Mulitmedia geographical Inforrmation Systems. *Innovations in GIS* 4, Taylor and Francis, pp. 56-69.

Mallat S. G. (1989) A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 11, No. 7, pp. 674-693.

Meer P., E. Baugher and A. Rosenfeld (1987) Frequency Domain Analysis of Image Pyramid Generating Kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 9, No. 4, pp. 512-522.

Montanari A. and B. Percini (1993) Temporal reasoning. In: A. Tansel, J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass (Eds.) , *Temporal Databases: Theory, Design, and Implementation. Database Systems and Applications Series*.Benjamin/Cummings, Redwood City, CA.

Montanari A., E. Maim, E. Ciapessoni, and E. Ratto (1992) Dealing with Time Granularity in the Event calculus. *Proceedings of the International Conference on Fifth Generation Computer Systems* 1992, Volume 2, Tokyo, Japan, ICOT, pp. 702-712.

Mountrakis G., P. Agouris and A. Stefanidis (2000) Navigating through Hierarchical Change Propagation in Spatio-Temporal Queries. *Proceedings of Seventh International Workshop on Temporal Representation and Reasoning (TIME 2000)*, Cape Breton, Nova Scotia, Canada, IEEE Press, pp. 123-13 1.

Niezette M. and J. Stevenne (1992) An efficient symbolic representation of periodic time. *Proceedings* of *CIKM*, Baltimore, pp. 161-168.

Ooi B. and R. Sacks-Davis (1989) Query Optimization in an Extended DBMS. In W. Litwin and H.J. Schek (Eds.), *Foundations of Data Organization and Algorithms,* Paris, Lecture Notes in Computer Science 367, Springer-Verlag, Berlin, pp. 48-*63.*

Ozsoyoglu G. and R. Snodgrass (1995) Temporal and Real-Time Databases: A Survey. *IEEE Transactions on Knowledge and Data Engineering,* 7, 4.

Peuquet D. (1984) A conceptual framework and comparison of spatial data models, *Cartographica,* 21 (4), pp. 66-l 13.

Peuquet D. and E. Wentz (1994) An Approach for Time-Based Spatial Analysis of Spatio-Temporal Data. *Advances in GIS Research,* Proceedings 1. pp. 489-504.

Quattrochi D. and M. Goodchild (eds.) (1997) *Scaling in Remote Sensing and GIS.* Boca Raton, FL, CRC/Lewis Publishers, Inc.

Rosenhohn D. (1987) Multi-point matching using the least-squares technique for evaluation of three-dimensional models, *Photogrammetric Engineering and Remote Sensing, (53) 6*, pp. 621-626.

Salzberg B. and V.J. Tsotras (1999) A Comparison of Access Methods for Time-Evolving Data. *ACM Computing Surveys.*

Sarda N. (1990) Extensions to SQL for historical databases. *IEEE Transactions in Knowledge and Data Engineering,* vol. 2, no. 2, pp. 220-230.

Schenk T., J.C. Li, and C. Toth (1991) Towards an autonomous system for orienting digital stereopairs. *Photogrammetric Engineering and Remote Sensing (57) 8,* pp. 1057-1064.

Segev A. and A. Shoshani (1993) A temporal data model based on time sequences. *Temporal Databases: Theory, Design, and Implementation.* Tansel et al., (eds.), The Benjamin /Cummings Publishing Company, pp. 248-270.

Sellis T. K. (1999) CHOROCHRONOS: Research on Spatiotemporal Database Systems. *DEXA Workshop 1999,* pp. 452-456.

Shahar Y. and M. Molina (1998) Knowledge-Based Spatiotemporal Linear Abstraction. *Pattern Analysis and Applications,* 1(2), pp. 91-104.

Shapiro L.G. and R.M. Haralick (1987) Relational matching. *Applied Optics, (26)* 10, pp. 1845-1851.

Shekhar S., S. Ravada, and X. Liu (1999) Spatial Databases-Accomplishments and Research Needs, *IEEE Transactions on Knowledge and Data Engineering,* Vol. 11, No. 1, pp. 45-54.

Silva F.L., L.B. Almeida, and J.C. Principe (Eds.) (1997) Spatiotemporal Models in Biological and Artificial Systems, *Frontiers in Artificial Intelligence and Applications,* Volume 37, pp. 210.

Sivapalan M. and J. D. Kalma (1995) Scale problems in hydrology: Contributions of the Robertson Workshop. *Hydrological Processes* 9(3/4), pp. 243-250.

Smith B. and D. Mark (1998) Ontology and Geographic Kinds. *Proceedings of International Symposium on Spatial Data Handling (SDH'98)*, Vancouver, Canada, 12-15 July, 1998.

Snodgrass R. (1987) The temporal query language TQUEL. *ACM Transactions on Database Systems,* vol. 12, no. 6, pp. 247-298.

Snodgrass R. (1992) Temporal databases. In: A. Frank, I. Campari, and U. Formentini (Eds.) *Theories and methods of spatio-temporal reasoning in geographic space.* Lecture Notes in Computer Science 639, Springer-Verlag, Pisa, Italy, pp. 22-64.

Snodgrass R. and I. Ahn (1985) A taxonomy of time in databases. In : S. Navathe (Ed.), *Proceedings of ACM SIGMOD International Conference on Management of Data,* Austin, TX, pp. 236-246.

Snodgrass R. T. (ed.), I. Ahn, G. Ariav, D.S. Batory, J. Clifford, C.E. Dyreson, R. Elmasri, F. Grandi, C. S. Jensen, W. Kafer, N. Kline, K. Kulkanri, T.Y.C. Leung, N. Lorentzos, J.F. Rod-dick,A. Segev, M.D. Soo, and S.M. Sripada (1995) *The Temporal Query Language TSQL2.* Kluwer, Academic Pub.

Story P.A. and M.F. Worboys (1995) A Design Support Environment for Spatio-Temporal Database Applications. *Proceedings of the International Conference on Spatial Information Theory (COSIT'95)*, Lecture Notes in Computer Science, Vol. 988. Berlin, Springer-Verlag.

Suetens P., P. Fua and A.J. Hanson (1992) Computational Strategies for Object Recognition, *ACM Computing Surveys,* Vol. 24, No. 1, pp. 5-6 1.

Tanimoto S. and T. Pavlidis (1975) A Hierarchical Strucutre for Picture Processing. *Computer Vision, Graphics and Image Processing,* Vol. 4, No. 2, pp. 104-l 19.

Tansel A., J. Clifford, S. Gadia, S. Jajodia, A. Segev, and R. Snodgrass (eds.) (1993) *Temporal Databases: Theory, Design, and Implementation. Database Systems and Applications Series.* Benjamin/Cummings, Redwood City, CA.

Tryfona N. and C. S. Jensen (1999) Conceptual Modeling for Spatiotemporal Applications. *Geoinformatica,* Vol. 3, No. 3, pp. 245-268.

Tsotras V. J. and A. Kumar (1996) Temporal database bibliography update. *SIGMODRecord,*25(1):11.

Tsotras V.J., C.S. Jensen, and R.T. Snodgrass (1998) A Notation for Spatiotemporal Queries, *ACM Sigmod Record.*

Vosselman G. (1992) Relational matching. *Lecture Notes in Computer Science 628,* Springer, Berlin.

Vosselman G. and H. Veldhuis (1999) Mapping by dragging and fitting of Wire-Frame models, *Photogrammetric Engineering and Remote Sensing,* Vol. 65, No. 7, pp. 769- *776.*

Wang X., S. Jajodia, and V. Subrahmanian (1993) Temporal Modules: An Approach Toward Temporal Databases. *Proceedings of SIGMOD Conference,* pp. 227-236.

Wiederhold G., S. Jajodia, and W. Litwin (1991) Dealing with Granularity of Time in Temporal Databases. *Proceedings of 3rd Nordic Conference on Advanced Information Systems Engineering,* Trondheim, Norway.

Witkin A. P. (1983) Scale-Space Filtering. *IJCAI* 1983: 1019-1022.

Witkin A. P. , D. Terzopoulos, M. Kass (1986) Signal Matching Through Scale Space. *AAAI,* Vol. 1 1986: 714-719.

Wong D. and C. Amrhein (eds.) (1996) The Modifiable Area1 Unit Problem. *Special issue of Geographical Systems 3* :2-3.

Worboys M. (1994) A Unified Model for Spatial and Temporal Information. *The Computer Journal,* 37,1.

Worboys M. (1995) *GIS: A computing perspective,* Taylor and Francis, London, UK.

Worboys M. (1992) Reasoning about GIS Using Temporal and Dynamic Logics. *NCGIA Technical Report, 90-4.*

Yacoob Y. and M. J. Black (1999) Parameterized modeling and recognition of activities. *Computer Vision and Image Understanding,* to appear.

Yip H.M. and T.C. Pong (1996) Detection of Moving Object in a Spatiotemporal Volume, *ICPR96,* (A80.27).

Yuan M. (1994) Wildfire conceptual modeling for building GIS space-time models. *GIS/LIS'94*, pp. 860-869.

Zheng Q. and R. Chellappa (1995) Automatic Feature Point Extraction and Tracking in Image Sequences for Arbitrary Camera Motion. *International Journal of Computer Vision,* 15:3 l-76.

Biography of the Author

Giorgos Mountrakis was born in Heraklion, Greece. He attended school in Heraklion, graduating from the 5th Kapetanakion High School. Directly after he started attending the National Technical University of Athens, in the Rural and Surveying Engineering Department. His research interests at that time were related to close range photogrammetric applications. He obtained his Diploma in Engineering in 1998, after defending his thesis "Determination and Correction of Radial Distortion in super-wide angle lenses of non-metric cameras': Following a working experience in Athens for the second quarter of 1998, he moved to Maine to pursue a Master's degree. In September 1998 Giorgos began his studies and served as a Graduate Research Assistant in the Department of Spatial Information Science and Engineering. As a graduate student, he published papers in several refereed conference proceedings and a journal. Since 1998, he is a certified engineer through the Technical Chamber of Greece. He is also a student member of SPIE.

Giorgos Mountrakis is a candidate for the Master of Science degree in Spatial Information Science and Engineering from the University of Maine in August, 2000.