TOWARDS AUTOMATIC MODELING OF BUILDINGS IN INFORMAL SETTLEMENTS FROM AERIAL PHOTOGRAPHS USING DEFORMABLE ACTIVE CONTOUR MODELS (SNAKES)

By

Hagai Mbakize Martine

Thesis Presented for the Degree of
DOCTOR OF PHILOSOPHY

Department of Geomatics
Faculty of Engineering and The Built Environment
University of Cape Town
South Africa

June 2001

" WE LIVE ON AN ISLAND OF KNOWLEDGE SORROUNDED BY A SEA OF IGNORANCE, AS OUR ISLAND GROWS SO DOES THE SHORE OF IGNORANCE "

(John A. Wheeler)

DECLARATION

I hereby declare that this thesis is my original work and that has not been submitted in any form to another university.

Hagai Mbakize Martine

ABSTRACT

This dissertation presents a novel system for the semi-automatic modeling of buildings in informal settlement areas from aerial photographs. The building extraction strategy is developed and implemented with the aim of generating a desk top Informal Settlement Geographic Information System (ISGIS) using self developed and available PC-based GIS tools to serve novice users in informal settlement areas. The proposed system uses a novel strategy of delineating buildings by optimizing their approximate contour positions which are derived automatically from digital surface models as elevation blobs. Building extraction is effected using the snakes paradigm and dynamic programming optimization technique. The proposed system, though semi-automatic, has most of its processes automated.

The need for the development of ISGIS is the attributed to rapid increase of informal settlement areas in developing countries, which has consequently lead to the degradation of the local ecosystems e.g. soil erosion, poor water, poor sanitation and severe social problems. Up-to- date, spatial data is one of the pre-requisites for improving living conditions in informal settlement areas, as, for example, in the implementation of re-settlement programmes. Existing spatial modeling techniques are expensive in terms of equipment and expertise, labour intensive, time consuming and are ineffective in coping with highly dynamic environments typically found in informal settlement areas. It is therefore imperative to develop new spatial modeling techniques better suited to the low cost and low technology of developing countries. This thesis is a step towards the

development of an effective spatial data acquisition method for semi-automated modeling of buildings in informal settlement areas.

The proposed method follows a top-down control strategy. It is a model based approach in which object extraction is achieved by using contextual object knowledge, global considerations and constraints. The main steps of the proposed system are detection and reconstruction, which are realized through the following sequence of activities:

- Hypothesis generation (raised structures hypotheses) from a digital surface model
- Location of raised structure hypotheses in orthoimages
- Building hypotheses verification
- Generation of approximate building contours for each building hypothesis
- Formulation of approximate building contours into snakes and optimization using the dynamic programming technique.

Inputs into the system are orthoimages and digital surface models. Digital surface models are generated by the image matching technique. Raised structures are modeled as "lumps" in the digital surface models. Lumps in 2-D are presented as elevation blobs, which are derived from the digital, surface models by altimetric segmentation. Initial windows for building extraction are provided by projecting centre points of the elevation blobs into corresponding orthoimages. Buildings are extracted from orthoimages using region growing constrained by edges, snakes and dynamic programming optimization techniques. A lump in a digital surface model can have causes other than a building, e.g. a tree, a car, a mound or other objects. To reduce the false alarm rate in building detection, it is thus imperative to perform building hypothesis verification so as to distinguish building hypotheses from other object hypotheses. In this research, building hypothesis verification is based on the analysis of frequencies of edge direction orthoimage patch's histograms. An orthoimage patch is an orthoimage area covered by an elevation blob. Resulting building hypotheses in vector format are used as initial or approximate building contours, which are formulated into snakes and input into the dynamic programming optimization process. The snakes and dynamic programming optimization process aim at modifying approximate building contours so as to generate new contours, which optimally delineate the buildings.

Central to dynamic programming optimization is the generic object mathematical model, which describes the manifestation of an object (in this case a building) in digital images. In this research, the generic building mathematical model used is based on the fact that building edges in an image have relatively higher edge magnitudes, and that pixels within a building region are presumably relatively homogeneous. The dynamic programming process is applied to each building's approximate contour after it has been formulated into snakes or active contour models, and from this process an optimal building contour is derived.

The proposed method was tested on two sites, which have different situations. The first study area was Manzese in Dar-Es-Salaam - Tanzania (East Africa), two sites were tested in this area. The second study area was Marconi Beam in Cape Town - South Africa, where one site was tested. 2-D building contours were extracted and 3-D building models were generated using the Arc View 3-D Analyst software. The results are encouraging, but not all buildings in the study areas were extracted. Buildings that were not extracted need to be by other extraction strategies and added later either manually or otherwise. This implies that the application of multi-cue algorithms is important in building extraction as no single algorithm to-date can correctly detect and delineate buildings in every scene, identical to what the human operator can perceive.

Despite building contour shape modification through the optimization process, the final extracted 2-D building contours were somewhat distorted. Interactive post editing of the 2-D building extraction output is thus necessary. The proposed system's performance was evaluated and a building detection rate of above 60% was realized.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my main supervisor Prof. Dr. Heinz Rüther, Head of the Department of Geomatics of the University of Cape Town, for his courage, guidance, criticism and challenges in doing this research. I would also like to thank my second supervisor Dr. E.G. Mtalo of the University College of Lands and Architectural Studies, a Constituent College of the University of Dar-es-Salaam in Tanzania. His corrections, criticism and comments helped me in shaping my work.

Without the assistance of staff members and students of the Geomatics Department of the University of Cape Town, this work wouldn't have been possible. Particularly, I would like to thank Andrea Court, Dr. Ulrike Rivett, my long time colleague and office mate Mbinji Mufalo, Sue Binedell, Nick Lindenberg, Prof. Mike Barry, Sidney Smith and Michael Haywood. I would also like to thank Dr. Julian Smith of the Emoyeni Training Centre for allowing me to use their workstation to generate and edit digital surface models.

It was only with the financial support from USHEPIA programme of the University of Cape Town that this research has been made possible. In particular I would like to thank Nan, Caz, Leslie and Carol for their positive outlook and support.

Last but not least I would like to thank my wife, Stella, for bearing the burden of looking after our family single-handed while I was away. Her emails always inspired me.

LIST OF FIGURES

Figure Page

LIST OF TABLES

LIST OF ACRONYMNS

The following acronyms are used in this thesis:

2-D	Two Dimensional
2.5D	Two and a Half Dimensional
3-D	Three Dimensional
DCS	Digital Camera System
DPW	Digital Photogrammetric Workstation
DSM	Digital Surface Model
DTM	Digital Terrain Model
GIS	Geographic Information System
ISGIS	Informal Settlement Geographic Information system
TIN	Triangular Irregular Network
UCT	University of Cape Town
UCLAS	University College of Lands and Architectural Studies
UNCHS	United Nations Centre for Human Settlements
UNDP	United Nations Development Planning
VRML	Virtual Reality Modeling Language

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1:1 CONTEXT

Effective management of human settlements is currently recognized as an important requisite for environmentally sound and sustainable management of development. Accurate and up-to-date information on human settlements is critical in fulfilling this requisite. At the same time the world is vulnerable to environmental impacts like atmospheric pollution, unpredictable climatic changes, floods and unsafe urbanization. However, access to Geographical Information Systems (referred to as GIS) technology which provides tools for sustainable management of the environment, natural resources and human settlements offers a breakthrough.

Data inputs for a GIS are commonly provided by ground surveying, conventional photogrammetry, existing base maps, and to a lesser extent, remote sensing. Due to low spatial resolution the use of remotely sensed data in GIS has been limited to small scale applications like derivation of thematic information, such as production of land use maps and change monitoring. However, advancements in digital sensor technology, digital image analysis techniques as well as computer software and hardware have resulted in the development of high spatial resolution sensors e.g. the 1m (panchromatic) and 4m (multispectral) of Ikonos (Ikonos, 1999). This development will subsequently open possibilities of using remotely sensed data in large scale applications for input into GIS. Digital photogrammetry has also enjoyed these technological advancements particularly from high-resolution hardcopy scanners and small format digital cameras as tested by Mason, Rüther and Smit (1997) and will soon benefit from airborne digital cameras (GeoEurope, 2000). Remote Sensing and Digital photogrammetry are now accepted as a primary source of data for GIS. They have the potential of lowering costs of data acquisition and analysis. In addition they meet other essential requisites, such as timely acquisition, accuracy and capability of synoptic area coverage (Burrough, 1992; Aronoff, 1991; Trotter, 1991; Gairns and Taylor, 1992; Mason, Rüther and Smit, 1997). Despite these technological advances, visual image analysis has remained more accurate and reliable and therefore superior to digital image analysis (Mulder, 1990; Trotter, 1991). At the same time, there has been a quest to automate GIS data capture methods particularly from Remote Sensing and Digital Photogrammetry data so as to optimally exploit the potential of digital images.

To date, digital image analysis, particularly automatic feature extraction, is far from being fully realized (Firestone et al, 1996; Gruen and Li, 1997; Schenk, 2000; Gülch, 2000). Nevertheless, semi-automated techniques have proved desirable in reducing the operational turn around time of providing data for GIS (Gairns and Taylor, 1992; Gruen and Li, 1997). The objective of automated data capture is access to a more effective system for data analysis and decision making.

The effectiveness of GIS's, also referred to as decision support systems, depends on the quality of the input data. These data need to be accurate and up-to-date so that decisions made using the data are optimal. A typical example of a need of information for decision making is in urban management, particularly in developing countries where cities grow rapidly and often this growth results in increased poverty, homelessness, environmental degradation and an inadequate supply of appropriate urban services. In such circumstances policy makers and urban planners need spatial information to monitor the impact of this growth, to evaluate existing urban policies and in turn to develop appropriate responses or strategies (UNCHS, 1997). The formation of informal settlements (generally referenced to as squatter areas) often goes alongside the growth of cities. Informal settlements are a common feature of developing countries and are typically the product of an urgent need for shelter by the urban lower income earners (Mason and Fraser, 1998). Their evolution in urban areas is an inevitable phenomenon. As long as urban areas offer economies of scale and agglomeration economies, large cities will always continue to grow, attracting migrants from rural areas and smaller urban areas, leading to more squatting. The United Nations forecasts that 40 percent of the world's urban population will live in unplanned areas by the year 2010 (Geo-information Africa, 1998).

In informal settlements areas, development in terms of spatial expansion and the right of occupancy are usually not in compliance with legal, urban and environmental standards set by governments. As a result these areas are prone to poor water quality and sanitation, and to severe social problems. Squatter upgrading is a measure taken to improve the quality of life of residents in these areas through a process of "regularization" whereby at least a form of secure tenure is established. Additionally, in the process of squatter upgrading, access to the urban infrastructure and services is enabled and housing as well as the physical and social infrastructure are upgraded to comply with minimum standards. Effective informal settlement upgrading requires reliable spatial data and application of spatial information technology in making decisions. Examples of tasks requiring spatial data include the monitoring of settlement growth, the re-allocation of residents to formal housing, the provision of a basic infrastructure, management and assessment of disasters e.g. floods, fire and adverse impacts on the local environment. The acquisition of spatial data involves the application of spatial information technology.

For spatial technology to be effective in the informal settlements environment, it has to be low cost both in data acquisition and processing, semi-automated or automated (to achieve faster and more reliable results), simple to use by low-skilled operators and as far as possible based on off-the-shelf software components e.g. a desktop GIS. Contrary to these requirements the acquisition of spatial data in informal settlements to-date has been based on conventional mapping techniques and mostly conventional photogrammetry using either analogue or analytical techniques. These techniques are mostly manual, in addition to that they require high technical expertise and expensive equipment. Therefore, for relatively small, densely populated areas and rapid changing environments typical of informal settlements these techniques do not suffice. That is, the techniques can not be easily applied on a regular basis (Li, Mason and Rüther, 1998). Alternative mapping techniques are therefore needed.

The objective in this research was to develop a low cost semi-automatic method of building extraction in informal settlements from high spatial resolution aerial images. This was done by using digital surface models as cues in the derivation of building hypotheses followed by the generation of approximate building contours using the region growing constrained by edges approach, the formulation of active contour models (snakes) and optimization using the dynamic programming technique. This research is a contribution towards spatial data acquisition methodologies for the support of the development of a low cost Informal Settlement Geographical Information System (ISGIS) using available GIS facilities together with self developed tools.

1:2 MOTIVATION AND RATIONALE

The research into semi-automatic building extraction in informal settlement areas from high-resolution aerial images was motivated by five issues:

1. Most research concerning building extraction has concentrated mainly on buildings in planned or formal settlement areas of developed countries. Little research has been dedicated to the extraction of buildings in developing countries particularly in informal settlement areas.

2. The need to acquire, manage and disseminate data to enable effective planning to address the problems of informal settlements as described in § 1:1.

3. Most of the existing systems of building extraction apply shadows in the generation of hypotheses and/or the estimate of heights of buildings. Geometric hypotheses, e.g. a roof should have 90° corners, or opposite edges of a roof should be parallel and so on, are taken into account in many existing systems. Images used in most of these systems are generally simple i.e. with few details and low density of buildings. Simple images generally require simple algorithms (Gruen, 1996). Tests have shown that these systems are unsuitable in situations where images are complicated as in informal settlements areas where there are no explicit or complete shadows, roofs are complex and irregular and the density of buildings is relatively high.

4. Automatic feature extraction is an area where digital photogrammetry seemed it could provide an added benefit to analytical techniques (Walker, 1997). It is expected that it could reduce fatigue and the work load of photogrammetrists. This expectation has not been fulfilled to-date and more research is therefore needed in this area.

- With the rapid advance of Earth Observation Systems technology, there is a need for developing tools for interpretation, handling of large amounts of imagining data and the extraction of feature information in as little time as possible.

1:3. PROBLEM DEFINITION

1:3:1 FEATURE EXTRACTION

Feature extraction is defined by Erik de Man (1988) as a process of getting, deriving, deducting a certain amount of information from appropriate sources, mostly sensors, to gain knowledge on position of objects or phenomena on the earth, their attributes and topological relationships. In Geomatics, feature extraction is defined as a process of detection and delineation of an object's spatial extent i.e. the precise location of object boundaries. It is essentially a process of spatial and non spatial feature data acquisition from sensor data. The process involves the extraction of relevant feature information by suppressing or neglecting other information.

Feature data is required mainly for applications like mapping for general planning, resource assessment and management, monitoring and as inputs into GIS's for integration with other data sets. Data integration is critical in GIS's because it facilitates the analysis of different scenarios for the purpose of predicting future developments, sustainable resource management and so on.

Many methods of feature extraction exist, however, the traditional method for obtaining accurate 3-D data of the terrain and man-made features has been conventional photogrammetry. Photogrammetry is a well-established and effective technique but some of its operations are still manual, labour-intensive and time consuming. Photogrammetry involves expensive equipment as well as specialized expertise. For rapidly changing spatial environments, as those found in informal settlements, the method becomes relatively slow. Alternative rapid spatial data acquisition techniques using satellite images and scanned aerial photographs have been sought and developed by several research groups for the past twenty years (Huertas and Nevatia, 1988; Liow and Pavlidis. 1990; Mckeown, 1990; Weidner and Förster, 1995; Firestone et al, 1996; Gruen and Li, 1997; Mason and Rüther, 1998).

High resolution images (aerial photographs and space borne satellite data) in combination with recent advances in sensor, software and hardware technology have greatly influenced the processes of feature extraction to be realized in automated environments. It is important and desirable to capitalize further on the potential of digital images as a primary source of data for GIS through the development of schemes involving automatic data acquisition methods. The key to improving GIS efficiency is the introduction of automation in the feature extraction processes.

A significant amount of work which has been done in the field of feature extraction has mainly concentrated on the extraction of man-made features, particularly roads and buildings. This is because buildings and roads are important, also they are predominant and frequently occurring man-made structures. Road extraction techniques are rather well established because roads can be described by generic mathematical models and their characteristics in images are known from our (human) knowledge about them. For example, a road pixel in the image generally has a higher intensity than its neighbour pixels on either side of the road. In the other words, in a digital image, a road can easily be identified as a narrow region of high intensity with lower intensity on both sides. At the same time, the contrast along the road usually does not change significantly within short distances. This stems from the fact that within short distances road materials typically do not vary much, and therefore, their spectral properties are almost constant. Hence, well-established algorithms are in place for road extraction, for example, road extraction by using dynamic programming and least squares B-Splines (Gruen and Li, 1997) and knowledge based road extraction from aerial images (Trinder et al, 1997).

1:3:2 BUILDING EXTRACTION

1:3:2:1 GENERAL

There has been significant research work in building extraction in planned or formal settlement areas of developed countries (Nicolin and Gambler, 1987; Huertas and Nevatia, 1988; Mohani and Nevatia, 1989; Liow and Pavlidis, 1990; Haala and Hahn, 1995; Weidner and Förstner, 1990; Haala and Brenner, 1997; Henricsson and Baltsavias, 1997; Paparoditis, et al, 1998; Seresht and Aziz, 2000). Little research has however been done towards the extraction of buildings in informal settlement areas (Mason and Rüther, 1997; Mason and Baltsavias, 1997; Li, Mason and Rüther, 1998; Mason and Fraser, 1998; Li, 1998). Despite all these research efforts, no single technique to-date has evolved as a solution to the building extraction problem. This is attributed to the complexity of buildings particularly in informal settlement areas where buildings are made up of diverse building materials and have varying shapes and orientations. Buildings in informal settlements do not follow regular codes but are subject to a complex interplay of natural, social, and political forces. Their extraction is thus difficult. It must be noted that modeling requirements for buildings differ between informal and formal urban settlements. Building extraction in informal settlement areas is a special challenge to photogrammetry, computer vision, and image understanding communities. Almost all building extraction systems developed to-date contain a problem-specific control structure. Therefore the adaptation of these systems to even slightly changed conditions or new applications remains difficult (ISPRS Highlights, 1998, Schenk, 2000). More research is thus still needed to explore possible avenues of building extraction, particularly in informal settlement areas.

The problem to be solved cannot be well defined without briefly highlighting some of the problems of existing methods of building extraction. Current building extraction strategies may be grouped into two main categories:

- 2-D building extraction i.e. boundary detection and delineation and
- 3-D building extraction i.e. modeling.

1:3:2:2 2-D BUILDING EXTRACTION

The 2-D building extraction category involves building hypotheses generation using low level single image analysis techniques, mostly shadows, edge and/or corner detection followed by edge linking and grouping (Liow and Pavlidis, 1990; Levine and Shaheen, 1981; Mckeown, 1984; Mckeown, 1990). Some approaches in this category use region image segmentation in the derivation of building hypotheses followed by verification using shadow analysis (Levine and Shaheen, 1981; Nazif and Levine, 1984). Other approaches, apply the snakes technique in which building seed points (mostly corners) are identified either manually or automatically from a database, followed by boundary position optimization through dynamic programming strategy. 2-D building extraction approaches mainly result in detection and object contour delineation in the image space. Such results though less accurate, are useful in general image interpretation and monitoring as in, for example, change detection. However, the approach is prone to the following pitfalls:

- Missing edges due to the failure of the edge extraction strategies in distinguishing between building and texture edges. Thus not all building lines are presented as edges.
- Edge fragmentation. In most cases an object will be represented by a number of disconnected edge fragments. Unless it is possible to associate or aggregate these fragments, they may be interpreted by the system as belonging to separate distinct objects. Fragmentation prevents the recognition of correct object properties thereby resulting in the inability to extract objects (Jaynes et al, 1994).
- Corner problems, which are due to poor contrast, result in gaps along edges causing difficulty in definition of corner points.

For the approaches based on hypothesis generation using image segmentation strategies, the image is initially segmented into regions, a region being a group of connected pixels with similar properties. These approaches are affected by the occurrence of artifacts, which are a result of either low contrast, between both the buildings and the background, or buildings with other adjacent buildings and features. Both edge/corner and region segmentation based building extraction strategies are low level image analysis techniques and as such do not deploy domain knowledge in processing. As a result, they perform poorly, particularly in images with poor contrast and high density of buildings, as those of informal settlements areas.

Some approaches to the 2-D building extraction category apply the technique of combining image primitives or tokens i.e. edges, corners and regions or segments similar to the method explained above, but in this case using apriori knowledge of building models expected in the scene. Such approaches are referred to as Knowledge Based Systems (KBS) (Nicolin and Gabler, 1987; Nagao and Matsuyama, 1980; Nazif and Levine, 1984). Knowledge Based Systems generally rely on the generation of object models using knowledge about the buildings expected in the scene, and compares the object models with the structural configuration of primitives as extracted from the image. Most of these systems are rule based i.e. they use rules in the comparison process. Knowledge is used in these systems to constrain the search space (Mckeown and Mcdermontt, 1985; Matsuyama, 1987).

1:3:2:3 3-D BUILDING EXTRACTION

3-D building extraction involves the analysis of digital images mostly by image matching to determine conjugate points from stereo or multi-images (Baltsavias, 1991; Henricsson and Baltsavias, 1997). The output of stereo analyses are point positions. Image matching is currently a standard tool for obtaining a 3-D model of the terrain and man-made features. The result of the image matching process is a Digital Height Model often referred to as a Digital Surface Model (referred to as DSM). DSMs, being graphical representations of objects, model both terrain and non-terrain features. Thus they can be used in the extraction of non-terrain objects like buildings, trees and other objects. The problems of using DSMs which are generated by the image matching technique in feature extraction include:

- In-sufficient ground sampling density to model discontinuities in-between buildings. Some buildings may even disappear.
- Height discontinuities are smoothed out.
- Matching errors that occur around buildings due to occlusions, poor contrast with surrounding features and negative shadow effects for example from overlapping or irregular shadows. As a result, break lines and building boundaries are not included in the resulting DSMs. This leads to the poor modeling of objects (Baltsavias, Mason and Stallman, 1995).

1:3:2:4 PROBLEMS IN BUILDING EXTRACTION

In general, the difficulties of building extraction include the negative effects of shadows, low contrast between buildings and the background, poor contrast in the roofs (roofs might be with textures or substructures). Additionally, non-fixed geometric shapes, interference by other objects (occlusions) for example, from trees and nearby buildings and distortions due to non-verticality of photographs limit the building extraction processes (Liows and Pavlidis, 1990; Huertas and Nevatia, 1988). For approaches which make use of shadows, in complex situations like those of the informal settlement areas where roofs are not flat, shapes of shadows may become irregular and consequently making the building extraction process difficult.

Most existing building extraction systems are semi-automatic. Full automation in feature extraction has been difficult to attain to date (Firestone, 1996; Gruen and Li, 1997; Gruen, 2000, Schenk, 2000). In view of that, semi-automated methods are a good alternative especially when one compares the speed and accuracy of a computer algorithm with the interpretation skills of a human operator. A semi-automatic system may be defined as a system which has a higher degree of automated processes than manual processes (Gülch, 2000). Indeed the system must be based on interactive decision-making coupled with quantitative input of parameters as needed.

1:3:3 STATEMENT OF THE PROBLEM

This research aims at developing a semi-automatic building extraction system that attempts to redress unsolved problems that limit the current building extraction techniques by applying the snakes and the dynamic programming optimization technique. It is reported by Gülch (1990) that snakes and dynamic programming optimization are effective and perform well even in the presence of photometrically weak boundaries, for example in poor contrast, edge fragmentation and so on. The proposed system is based on using contextual knowledge and cues to generate building hypotheses in orthoimages and optimizing building positions by the dynamic programming technique. As a pilot study, aerial photographs at a scale of 1:12,500 and focal length of 152mm scanned at a resolution of 15 microns and direct digitally acquired images using a small format camera Kodak DCS 460 are used.

1:4 ORGANISATION OF THE THESIS

The thesis is organized in eight chapters. Chapter One is a general introduction to the importance of GIS in the sustainable management of human settlements with the emphasis on informal settlement areas. The chapter highlights the need for automation in GIS data capture methods, reviews briefly problems of automation in contemporary building extraction systems and defines the problem to be solved by this thesis.

Chapter Two reviews building extraction strategies developed to date. Without going into detail, the chapter highlights the advantages and disadvantages of each strategy, with the aim of identifying remaining problem areas which form the objectives of the thesis.

Chapter Three briefly reviews the theoretical background of the tools used in designing the building extraction strategy proposed in this thesis. Image matching, snakes, dynamic programming, gray scale mathematical morphology and region growing techniques are covered. Chapter Four is a detailed description of the proposed building modeling strategy with some preliminary implementations. Chapter Five deals with important implementation issues along with the results and evaluation of the proposed building modeling system. Chapter Six presents the analysis of the results. Chapter Seven is a general discussion in terms of applicability and operationalization of the proposed system.

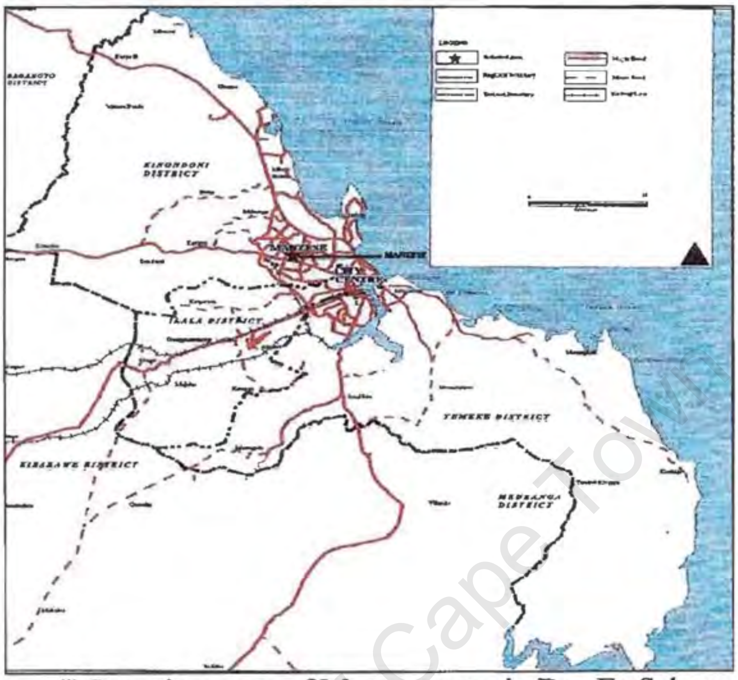
The thesis ends with Chapter Eight, which presents conclusions, recommendations and highlights on future research directions.

1:5 STUDY AREAS AND THEIR CHARACTERISTICS

For purposes of validating the proposed building extraction strategy, two study areas with differing characteristics, environments, geographical locations and data acquisition methodologies are used in this research.

The first study area of this research is part of Manzese(see Figures 1a and 1b) in Dar-es salaam city in Tanzania where estimation by the Dar-es-salaaam City Council indicates that approximately 70 percent of the population live in unplanned areas (UNDP report 1992). Manzese is one of the highly populated and fast growing suburbs of Dar-es-Salaam. The area is located about 10-km northwest of the city. According to the standards of the Tanzania Ministry of Lands and Human Settlements Development, the area is an informal settlement characterized by detached houses with almost uniform roof structures and varying geometric shapes. Most of the buildings are made up of concrete and the roofing is mainly corrugated iron sheets (refer Figure 1b). The area is relatively flat with little vegetation coverage. Residents of the area are mostly low-income workers in nearby industries and government offices; some are involved in self-established small businesses. The data for this area are from aerial photographs acquired in 1992 and scanned at a spatial resolution of 0.2 m.

The second study area is part of Marconi Beam, one of the informal settlements surrounding Cape Town in South Africa (see Figure 2). The area is characterized by informal buildings called shacks constructed largely of plastics, iron sheets and timber.



(i) Location map of Manzese area in Dar-Es Salaam

(ii) Image of Manzese Site1

(iii) Image of Manzese Site2

Figure 1a. Location map and images of the first study area-Manzese in Dar-Es-salaam, Tanzania.

(i)

(ii)

(iii)

(iv))

Figure 1b. Type of Buildings in the Manzese area.

(Photographs taken in December 1999)

(a) Location map of Marconi Beam Area in Cape Town

(b) Image of Marconi Beam Area

Figure 2. Location map and image of the second study area - Marconi Beam in South Africa.

CHAPTER 2

A REVIEW OF TECHNIQUES FOR BUILDING EXTRACTION

The purpose of this chapter is to provide a brief overview of building extraction methods developed to date with the emphasis on those which have greatly influenced this research.

2:1 FEATURE EXTRACTION

Although many researchers have made great strides in digital image analysis over the past twenty years, it is still a challenging task extracting significant man-made structures such as buildings, roads and other structures from aerial photographs or remotely sensed imagery,particularly in high-density urban areas (Nicolin and Gambler, 1987; Huertas and Nevatia, 1988; Mohani and Nevatia, 1989; Liow and Pavlidis, 1990; Haala and Hahn, 1995; Weidner and Förstner, 1990; Haala and Brenner, 1997; Henricsson and Baltsavias, 1997; Paparoditis, et al, 1998; Seresht and Aziz, 2000). Feature extraction started in the 1980's by Nagao and Matsuyama, who developed a fully automated system based on the so-called structural analysis techniques. Thereafter, many semi-automated approaches mainly for building extraction have been proposed and tested by several research groups.

2:2 GENERIC MODEL TECHNIQUES

Most building extraction approaches found in the literature use generic models to describe shapes of buildings expected in the scene. Earlier strategies of building extraction aimed at 2-D object boundary delineation in the image space. Fua and Hanson (1987) applied an image segmentation technique to extract buildings from an aerial imagery and by varying the homogeneity criterion; a result which had a maximal number of 90° building corners was picked as an optimal segmentation. The method is effective when the shapes of buildings are simple e.g. box shapes, and when good segmentation results are attained. Such conditions cannot be achieved in informal settlement areas. This is because buildings in informal settlement areas are made up of mixed building materials, a situation which limits the application of the image segmentation technique in building extraction. Nicolin and Gabler (1987) used a knowledge based system for the analysis of aerial images. Their system had four components: a knowledge base of domain independent processing techniques; a long term memory containing domain apriori knowledge; a short term memory containing intermediate results from the image analysis module and a control module responsible for invocation of various processing techniques.

In general, knowledge based image analysis systems involve three major processing steps:

1. Low level image processing involving mostly edges and corners detection to generate object primitives.
2. Grouping of object primitives to derive the object's primal sketches.
3. Feature extraction by deriving 3-D model representation of objects using context knowledge i.e. high-level image processing (Trinder, Wang and Parlang, 1997).

It can be envisaged that effectiveness of the knowledge based image analysis systems will mainly depend on the quality of the low level image processing results, from which object models are formed and subsequently compared with models in the knowledge base. In uncontrolled environments such as those in informal settlement areas, images will usually contain noise and ambiguities, which hinder the success of low level image operations. This then limits the application of knowledge based approaches in areas with high building density (Suetens et al, 1992).

Huertas and Nevatia (1988) developed a system of detecting buildings in aerial images using generic models of buildings expected in the image. Their system used line segments corresponding to intensity edges in the scene, edges were then thinned and linked to form continuous lines from which the object's primal sketches were derived. Shadows were used to confirm the presence of buildings and obtain an estimation of heights. The difficulties with this method include poor contrast i.e. buildings with poor contrast are missed altogether or resulting edges are fragmented; some buildings may be missed because of their small size and as the density of the buildings increases shadows tend to fall from one building onto another making edge extraction difficult.

Mohan and Nevatia (1989) applied a technique based on the detection of linear segments followed by perceptual organization which involved the grouping of fragmented low level descriptions into meaningful higher level object descriptions which in turn are used by higher level reasoning processes. In their technique, structural relations of image primitive were used to form the so-called collated features. The choice of collated features was determined by the generic shape of the desired objects expected in the scene. The output of the system was a 3-D description of objects. The effectiveness of this technique depends on contrast and the amount of detail in the image.

Liow and Pavlidis (1990) used integration of region growing and edge detection techniques in extracting buildings followed by verification using shadows. In their first method edge detection was used to locate shadowed boundaries. Then from the shadow boundaries, they applied region growing, together with binary erosion and dilation techniques, to find non-shadowed boundaries, so as to obtain a complete shape description of the buildings. In their second method, region growing controlled by edges was applied to obtain initial image segmentation. In both methods boundary artifacts were eliminated by applying a modification process. The end product was a 2-D building delineation in the image space. The building extraction method being proposed in this research draws building contour modification experiences from their work.

Other systems developed almost at the same time include the BABE system (built-up area building extraction) McKeown (1990), which analyzes intensity edges, estimates the shadow intensity and illumination direction and produces a set of building hypotheses. SHADE, SHAVE, and GROUPER were systems developed after the BABE system for shadow detection, verification and grouping of fragmented building hypotheses respectively by examining their output relationship to possible building/shadow edges. McKeown and Shufelt (1993) describe a method of integrating outputs of different feature extraction strategies to detect man-made features from aerial imageries i.e. a multi-cue algorithm approach. In their method, symbolic data generated by the SHADE, SHAVE, and GROUPER systems were fused resulting in an increased 2-D building detection rate. In efforts to alleviate some of the problems of monocular image analysis for example, segmentation artifacts, fragmented edges and so on, human intervention in image processing became necessary, and this led to the evolution of semi-automatic interactive object reconstruction techniques.

2:3 SEMI-AUTOMATIC TECHNIQUES

Semi-automatic techniques started in the early 90's after Kass et al (1988) introduced the concept of deformable active contour models or snakes. Snakes were used in interactive feature extraction environments (Gülch, 1990a and 1990b, Amini et al, 1990). In principle, the snakes approach involves the operator coarsely identifying seed points of a feature to be extracted, after which precise object boundary localization is achieved automatically by relaxation using energy optimization techniques. The solution is an optimal compromise between internal and external forces acting on the active contour model (refer § 3:2 for concepts of energy and energy optimization). The snakes approach has been used in the extraction of roof parts in digital aerial images (Gülch, 1995) (refer § 3:3 for a detailed description of the snakes method).

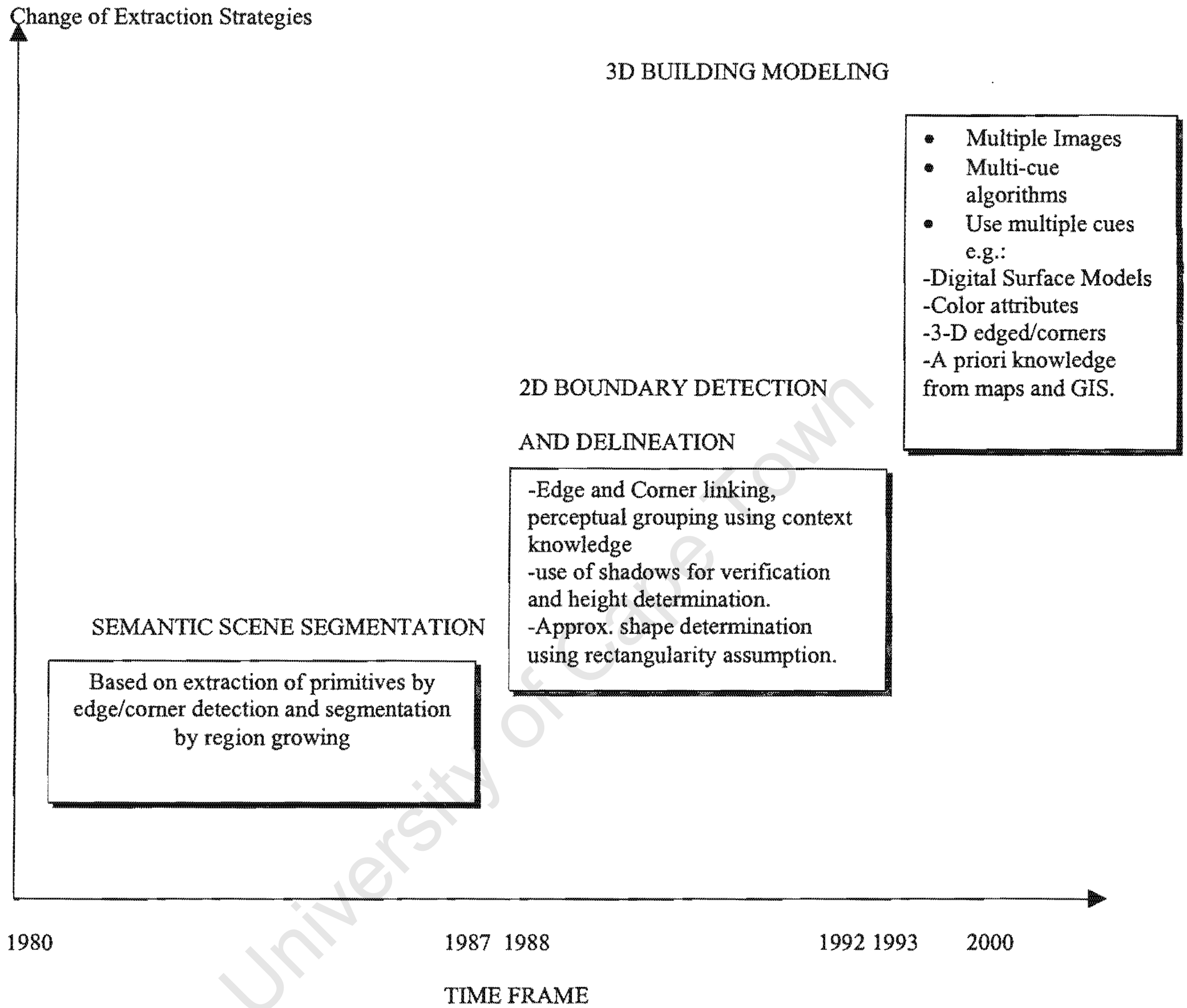


Figure 3. Evolution and shifting of building extraction techniques over time.

In the last few years, several (academic) groups have presented new promising results in semi automated 3-D building reconstruction (Haala and Hahn, 1995; Haala and Brenner, 1997; Weidner and Förstner, 1995; Paparoditis et al, 1998; Henricsson and Baltsavias, 1997; Henricsson, 1998). Research on shadow analysis as the main cue for inferring 3-D structures from monocular images has been on the decrease and has been slowly replaced by 3-D information such as digital surface models, 3-D edge and corner extraction from multiple overlapping aerial images (refer Figure 3).

In general, the change in the trend of building extraction techniques has been towards:

- The use of oblique image views
- Multi-image approaches
- Multi-cue algorithms
- Fusion of various information sources
- Digital Surface Models for detection and reconstruction
- Generic roof modeling by decomposition into parts
- Use of a priori knowledge from maps and GIS and
- Semi-automated reconstruction techniques (Grün, 2000).

This change of extraction techniques is attributed to advances in computer technology and the recent growing demand for 3-D city models for application in urban planning, architecture, telecommunication and environmental engineering. The automation of DSM generation has effectively attracted 3-D object reconstruction initiatives. The use of DSM in building extraction is motivated by the fact that it inherently provides a geometric description of the scene as derived from aerial imagery or airborne laser scanner data. Building extraction systems developed using DSMs include:

- The system by Weidner and Förstner (1995) which involved extracting 3-D building shapes from digital surface models (DSM) using constraints on building models. The DSM was computed using image and feature pyramids and the final surface was refined by means of local adaptive regularization techniques. This process resulted in a graph surface z = f(x, y). Building detection was then based on the fact that buildings are higher than the topographic surface, which was estimated using mathematical morphology of the DSM. Buildings were subsequently reconstructed depending on their complexity. For 3-D building reconstruction, parametric models were used for simple buildings, that is, those having either flat or symmetric sloped roofs while for complex or connected buildings, prismatic models were used. The method being proposed in this research draws 3-D building reconstruction experience from Weidner and Förstner's (1995) work, as most buildings in the study areas are single roofed and either symmetric or flat.

- The AMOBE project at the ETH Zürich Baltsavias, Mason and Stallman (1997), aimed at the semi-automatic 3-D reconstruction of man-made objects from aerial images. The data set used in the AMOBE project consisted of four vertical color aerial images having a ground resolution of about 10 cm with a precisely known sensor orientation. A DSM at a grid spacing of between 25 and 50 cm was generated using standard image matching algorithm. Elevation blob detection was performed on the DSM, and the results were combined with color image information in order to extract building hypotheses. Subsequent feature extraction processes involved edge detection, aggregation or grouping and computation of photometric attributes of edges. The proposed building extraction method draws experiences of using DSM in building detection from the AMOBE project.
- The ASCENDER system developed at the University of Massachusetts, Collins et al (1996), aimed at modeling rectilinear and flat roofs of isolated or connected

buildings. The data sets used in the ASCENDER system consisted of multiple high resolution aerial images with known exterior orientation. The best results were obtained with one nadir and four oblique i.e. E, W, N and S views, but combinations of other images were used as well. The first step in the ASCENDER system was the monocular detection of buildings by linear edge detection, corner extraction and perceptual grouping. False detection or gaps are corrected through multi-image analysis. Detected roof vertices are sought in all images and a 3-D wire frame representation of the scene is built up.

- The system by Roux and Mckeown (1994) which involved the reconstruction of 3-D roof surfaces of buildings by the integration of information from multiple view image analyses. The basis of the system was the detection of 3-D corners using two views and the extraction of polygonal surfaces from corner and edge cycles in a relational graph. This process involved the use of geometric constraints in the object space. In this system, analysis of additional images increased the 3-D accuracy of surfaces and added new buildings not detected in first views. The system was reported to be efficient when various viewpoints (nadir and oblique) were used.
- The ARUBA system developed by Henricsson and Baltsavias (1997) aimed at the automatic reconstruction of 3-D roof structure of a general class of buildings with a high metric accuracy from multiple high resolution overlapping color aerial images. The system was a collection of independent modules each charged with a specific feature extraction task, though, transfer of data and information between modules was possible. The main processing steps of the system were 2-D processing i.e. filtering; 2-D feature extraction; computing color region attributes; extracting similarity

relations and similarity grouping; 3-D processing i.e. stereo matching, coplanar grouping and assembly of 2-D enclosures and planes to complete buildings and interaction between 2-D and 3-D processing to resolve ambiguities. The algorithms, however, were not capable of fully automatic extraction of complex scenes like those of suburban and densely populated urban regions due to problems associated with the connection of buildings, shadows from trees and other buildings and objects occlusions.

- The system by Mason and Baltsavias (1997); Li, Mason and Rüther (1997), for semi-automatic shack extraction in which multiple cues such as edge contours, shadows and DSM were fused for shack detection and delineation. In their system, interactive manual support was provided during the extraction process to assist the automated procedure in the cases of complicated roofs and/or where cues were inadequate. The system ended up with refined blobs, which suffice for coarse delineation of individual shacks, and is adequate for many applications. The proposed method in this research draws experiences from Mason and Baltsavias's (1997) approach with the aim of making a further step towards precise informal buildings delineation.

Multispectral image classification is another strategy for man-made object extraction. The potential of multispectral classification in supporting man-made extraction in digital imagery is well recognized. However, todate it has found only limited application due to:

- The effect of spectral confusion between classes particularly in high resolution images. This effect is critical in urban areas and it is worse in informal settlement areas. In most informal settlements areas, the background is largely bare ground, which makes it spectrally similar to most building materials, thus rendering their extraction difficult (Mason and Baltsavias, 1995).

- The maximum likelihood model, which is the basis of multispectral classification, is based on the assumption that the object's reflectance characteristics obey the Gaussian normal distribution curve. This assumption is not always valid and may lead to mixed classes.
- The technique does not make use of global information, such as spatial relationships and objects context knowledge. Thus, image classification methods result in the characterization of spectral classes rather than object identification and description which are the core of standard image interpretation techniques (Demetre and Harlow, 1990).

Mason and Baltsavias (1995) attempted to use multispectral classification to generate shack hypotheses, and found this approach to be unacceptable due to the increased number of false alarm rates with most shacks being partially classified as bare ground. Though multispectral classification finds many applications in land use/ground cover assessment, it is not ideal for object extraction where accurate boundary delineation is important.

Most of the methods of building extraction developed todate are yet to be realized in operational commercial production lines on Digital Photogrammetric Workstations (referred to as DPW). This is because the performance of existing systems is still unacceptable for a commercial production environment. However, acceptable performances have been reported for isolated buildings (Gülch, 2000). In practice, few

DPW include automated feature extraction modules. On the Leica-Helava DPW, few tools for automated feature extraction are implemented. The most robust tool is in the current version of SOCET SET ® which enables the operator to place the measuring mark near the corners of buildings and the tool then finds the precise corners and completes the building (Walker, 1997). More research is therefore needed for the development of robust tools for automated feature extraction. According to Gülch (2000) any development of new automated tools should attempt to meet the following requirements:

- They must contain an acceptable degree of automation.
- They should be applicable to real world objects to be of practical use.
- They must be based mainly on the use of image data with possible integration of auxiliary data e.g. digital surface models.

The method proposed in this study addresses all the above requirements, at least partially.

2:5 SUMMARY

The overview above shows that early methods of building extraction and detection based their processing on single gray-valued images applying knowledge of objects, simple object models and shadow analysis. The main aim, then, was not to construct buildings in 3-D but to detect them and find their 2-D outlines in the image space. Most of the approaches relied on simple assumptions like the fact that buildings are characterized by significant geometric regularity e.g. rectangularity of building lines, rectilinear roofs, and so on. However, even in the assumption of such simple object models, analysis of monocular images is a difficult task since it generally leads to ambiguous solutions.

In general, weaknesses of the methods reviewed are poor contrast, density and the complexity of buildings. It is the objective of the proposed building extraction system to alleviate these weaknesses by adopting the snakes approach and the dynamic programming optimization strategy. The snakes approach incorporates the object's contextual knowledge into processing and replaces the pixel based methods by global considerations. The snakes method thus offers a link between low and high level image interpretation processes thereby bridging the gap between photometrically weak image points (Gülch, 1990a and 1990b).

In general, it can be concluded that a strategy of grouping of image primitives creating primal sketches of features is extremely important in building extraction. This is due to the fact that low level feature extraction alone cannot be expected to derive all object parts due to the problems of shadows, poor contrast and occlusions. Multiple views are a pre-requisite for many algorithms with color information being of increasing importance (Henricsson and Baltsavias, 1997). Most building extraction systems developed todate are non-generic, consequently the system's performance in slightly changed scene conditions is still poor.

CHAPTER 3

THEORETICAL BACKGROUND

This chapter discusses the theoretical basis of tools used in the proposed building extraction strategy. The theory and concepts of digital surface model generation by image matching and airborne laser scanning, active contour models (snake), dynamic programming, region growing, edge detection and gray scale morphological filtering will be covered.

3:1 DIGITAL SURFACE MODELS

In this research, digital surface models are used as cues in the generation of building hypotheses. There are two techniques of automatic digital surface model generation: image matching and airborne laser scanning. The digital surface models used in this research are generated by the image matching technique. Principles of image matching and airborne laser scanning are briefly explained in the following sections.

3:1:1 IMAGE MATCHING TECHNIQUE

Image matching is a process of locating corresponding or conjugate points of interest in stereo images for the purpose of three dimensional point determination in object space. Image matching techniques fall under three main categories, namely, area, feature and relational or structural based matching as shown in Figure 4. Area based matching determines the correspondence between two image areas according to the similarity of their gray values. Cross correlation and least squares correlations are common techniques of area based matching. Feature based matching determines the correspondence between image features.

In most cases feature based techniques match extracted point features as opposed to other features, such as lines and complex objects. The implementation of feature based matching hence involves initial image feature extraction. Förstner, Moravec, Canny and other operators are commonly applied in the extraction of interest points, which are subsequently used in matching, by comparing their attributes from overlapping images. (Förstner and Gülch, 1987, Canny, 1983, and 1986). Relational based matching uses the relations between features in addition to the image features. In relational based matching, corresponding image structures can be recognized automatically without apriori information (Ballard and Brown, 1992 Wang, 1998, Van der Merwe and Rüther, 1995). The scope of this research does not cover relational based matching, so no elaboration is given here. The generation of digital surface models as now made operational on digital photogrammetric workstations is usually realized through the integration of feature and area based matching techniques which is the subject matter of this section.

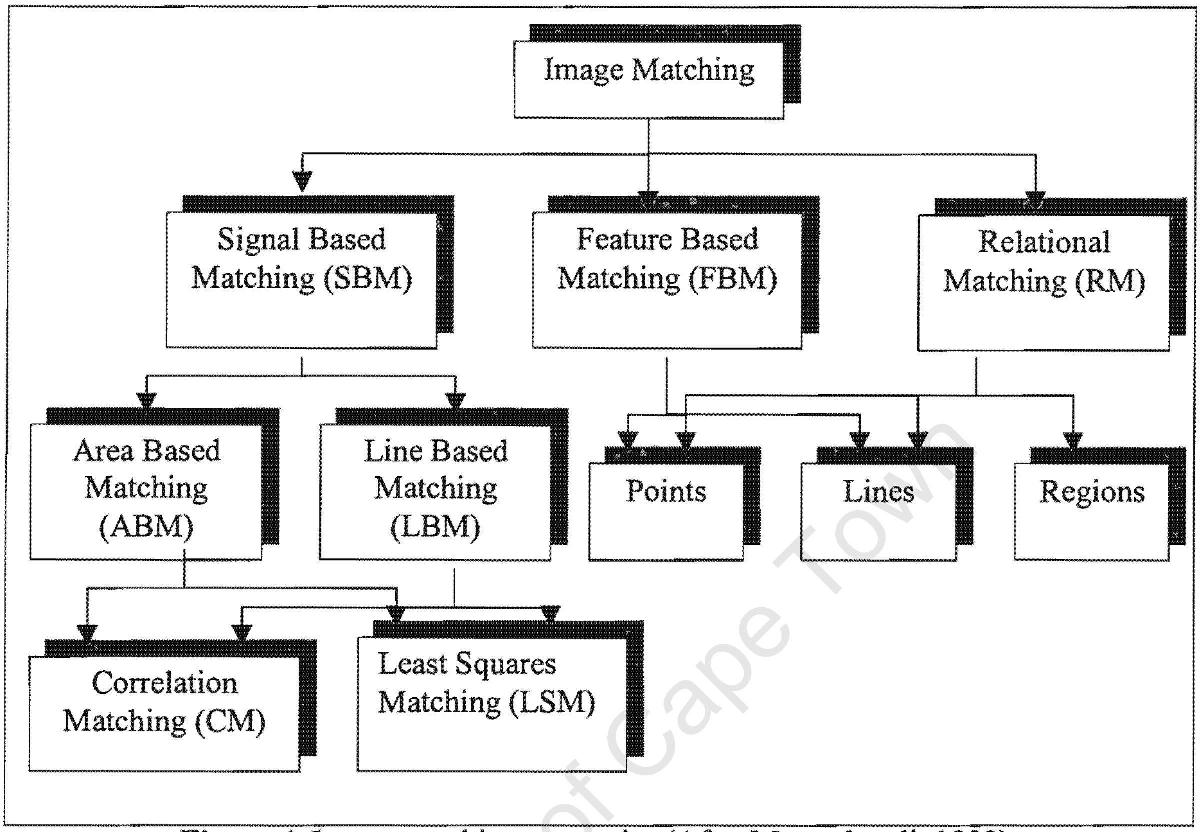


Figure 4. Image matching categories (After Msemakweli, 1999).

Area based matching can be done from either the image space or the object space (Msemakweli, 1999). The most common practice is working from image space and computing object space coordinates by space intersection. In the matching process, a priori knowledge of the search patch position in target images is usually employed to reduce the search space. Epipolar geometry is normally applied for this purpose (Baltsavias, 1991;Wong, 1996; Smith, 1997). An epipolar line is defined as a line of intersection between the image plane and the plane passing through the object point and the perspective centres of two images as shown in Figure 5. This plane also contains the image points of the object.

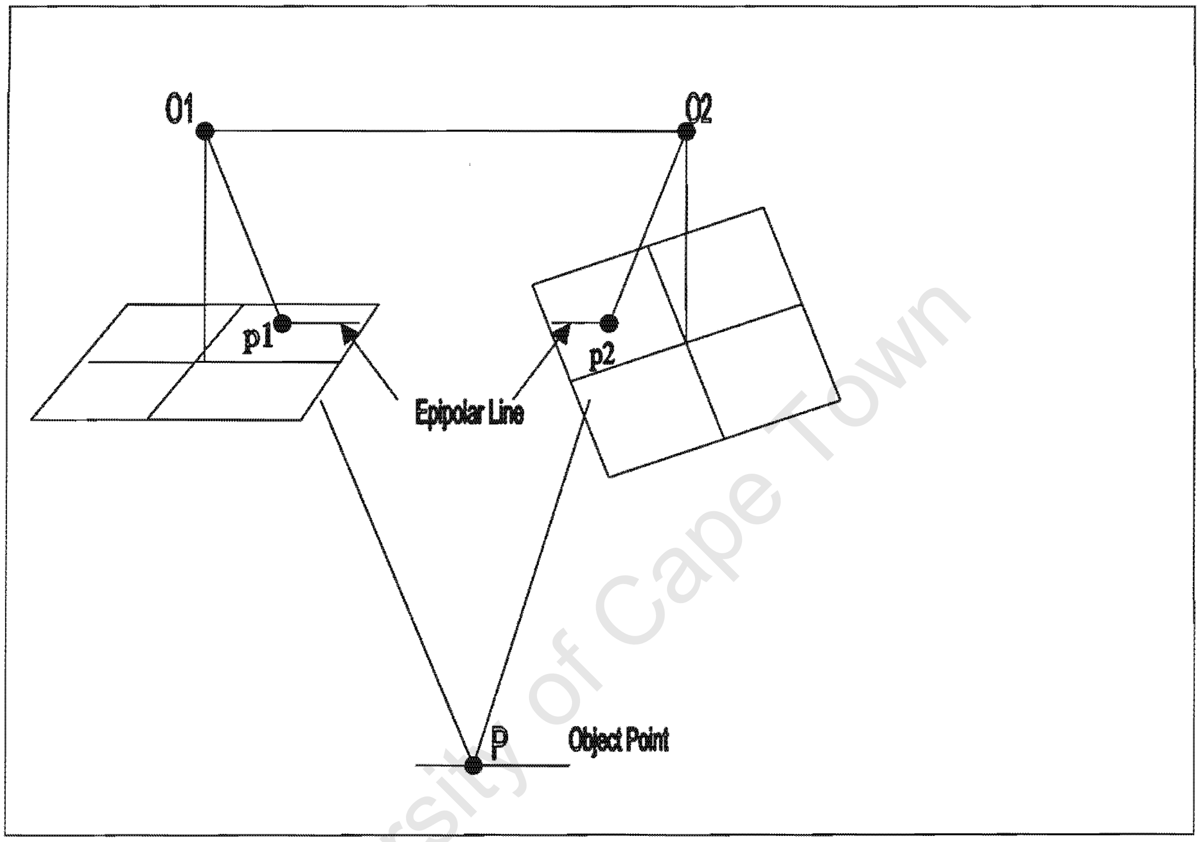


Figure 5. The concept of epipolar line.

3:1:1:1 MATHEMATICAL MODELS AND PRINCIPLES

The basic mathematical models of area based matching are cross correlation and least squares matching. Object space coordinates of interest points are obtained by space intersection using collinearity condition equations, which relate image points to corresponding object points in a perspective geometry (refer Equations 3:1). For point 'a' in an image, collinearity condition equations are defined as:

$$x_a = x_p - dx + f\left(\frac{r_{11}(X_A - X_C) + r_{12}(Y_A - Y_C) + r_{13}(Z_A - Z_C)}{r_{31}(X_A - X_C) + r_{32}(Y_A - Y_C) + r_{33}(Z_A - Z_C)}\right)$$

$$y_a = y_p - dy + f\left(\frac{r_{21}(X_A - X_C) + r_{22}(Y_A - Y_C) + r_{23}(Z_A - Z_C)}{r_{31}(X_A - X_C) + r_{32}(Y_A - Y_C) + r_{33}(Z_A - Z_C)}\right) \quad (3:1)$$

Where:

x_a, y_a - are image coordinates of a point of interest 'a'.

x_p,y_p and c are principal point coordinates and principal distance respectively.

dx, dy - are lens and other image distortions.

X_A,Y_A, Z_A - are object space coordinates of a ground point A

X_C,Y_C, Z_C - are perspective centre object space coordinates and

r_{11} - r_{33} - are elements of a rotation matrix.

f – Focal length of the camera

Inputs to the image matching process are overlapping digital photographs, distortion parameters and interior and exterior orientation parameters. It is important that prior to the matching process there are enough control points to allow for the accurate computation of absolute orientation parameters which are ω, φ, K, X_C, Y_C, and Z_C. where ω, K φ, , are rotational angles around object space X,Y,Z axes respectively and X_C, Y_C, Z_C are perspective center object space coordinates. Area based matching starts with the extraction of interest points in one of the input images called the reference image. A reference window defined at each interest point is used together with the search window defined in the target images to search for conjugate interest points in target images.

3:1:1:1:1 CROSS CORRELATION MATCHING

The process involves only two images at a time, the reference and the target image. Search along the target image as earlier mentioned is constrained along epipolar lines (see Figure 6). Epipolar lines positions are computed using collinearity and coplanarity condition equations. In most cases the search window will not match in shape with the reference window. This may be attributed to the following factors:

- Position and orientation of the target images with respect to reference image.
- The difference in height of the surface elements imaged within the reference window (Van der Vlugt, 1995).

To model these factors and other unknown effects of camera geometry the search window needs to be reshaped and resampled prior to the matching process. This is done by first transforming the reference window, using collinearity condition equations, into the image and later to the pixel coordinate system of the search window, and secondly the computation of affine transformation shaping parameters (refer Equations 3:2). This is followed by bilinear interpolation to determine gray values of a transformed reference window (Smit, 1997).

$$x = a_0 + a_1 x^o + a_2 y^o$$

$$y = b_0 + b_1 x^o + b_2 y^o \qquad (3:2)$$

Where:

a_o to a_2 and b_o to b_2 - affine unknown shaping parameters.

x^o and y^o - original unshaped pixel positions.

x, y - shaped pixel positions.

After computation of affine shaping parameters, for each pixel in the reference window a corresponding pixel in the target image is determined as well as its gray value. The correlation coefficient between the reference and the transformed search window is then computed using well known normalized cross correlation equations defined as:

$$NC = \frac{\sum (g_r - \overline{g_r})(g_s - \overline{g_s})}{\sqrt{\sum (g_r - \overline{g_r})^2}\sqrt{\sum (g_s - \overline{g_s})^2}} \qquad (3:3)$$

Where:

g_r and g_s - are gray scale values on reference and search windows respectively.

g_r and g_s - are mean gray scale values for reference and search windows respectively.

NC – normalized cross correlation coefficient.

In the matching process the search window moves to various position along the epipolar line. A search window with maximum normalized cross correlation coefficient value is chosen as the best matching window. Disadvantages of cross correlation matching includes its limitation to only two images and poor accuracy because it is based on gray level matching without consideration of geometry. However, it is used to provide provisional values for least squares matching algorithm as explained in the following section.

3:1:1:1:2 LEAST SQUARES MATCHING (MPGC)

This is Multi Photo Geometrically Constrained (referred to as MPGC) least squares matching. It is rigorous and thus capable of better point position accuracy and reliability than the conventional cross correlation matching. The MPGC approach allows the exploitation of any a priori known geometric information to constrain the solution and the use of more than two images. The MPGC approach is a currently a standard technique of area based matching. The mathematical models used in the MPGC approach are formulated in terms of the least squares model. The least squares observation equations are formulated to consist of two parts, i.e. the equations that express the gray level matching and the equations that express the geometric conditions that must be fulfilled . The two parts are related to each other forming a joint system of equations through the unknown parameters or through a function that appear in both set of equations. Otherwise the two parts are independent of each other and the geometric information does not influence the matching. Functional models for gray level matching and for geometric conditions between conjugate images are formulated and developed by Baltsavias, (1991) and Smit, (1997), leading to the least squares solution of the joint system being expressed as:

$$X = (A^T P A + B^T P_t B)^{-1} (A^T PL - B^T P_t t) \tag{3:4}$$

Where:

A – Design matrix as obtained by linearization of gray level matching observation equations,

P – Weight coefficient matrix.of linearized gray level observation equations,

P_t – Weight coefficient matrix of linearized collinearity condition equations,

B – Design matrix as obtained from linearization of collinearity condition equations,

L -Gray level equations observation vector and

t - Collinearity condition equations observation vector.

In the joint mathematical model (refer Equations 3:4) a simultaneous solution for the corresponding image points in conjugate target and reference images is provided, the unknowns are shift parameters for each search window in a target image and object space coordinates of the interest points. The final solution is obtained iteratively per interest point in which approximate values for the non-linear parameters (i.e. six exterior orientation parameters for each image and object space coordinates) are required as initial parameter values. The iterations are terminated when each element of the solution vector falls below a user defined threshold.

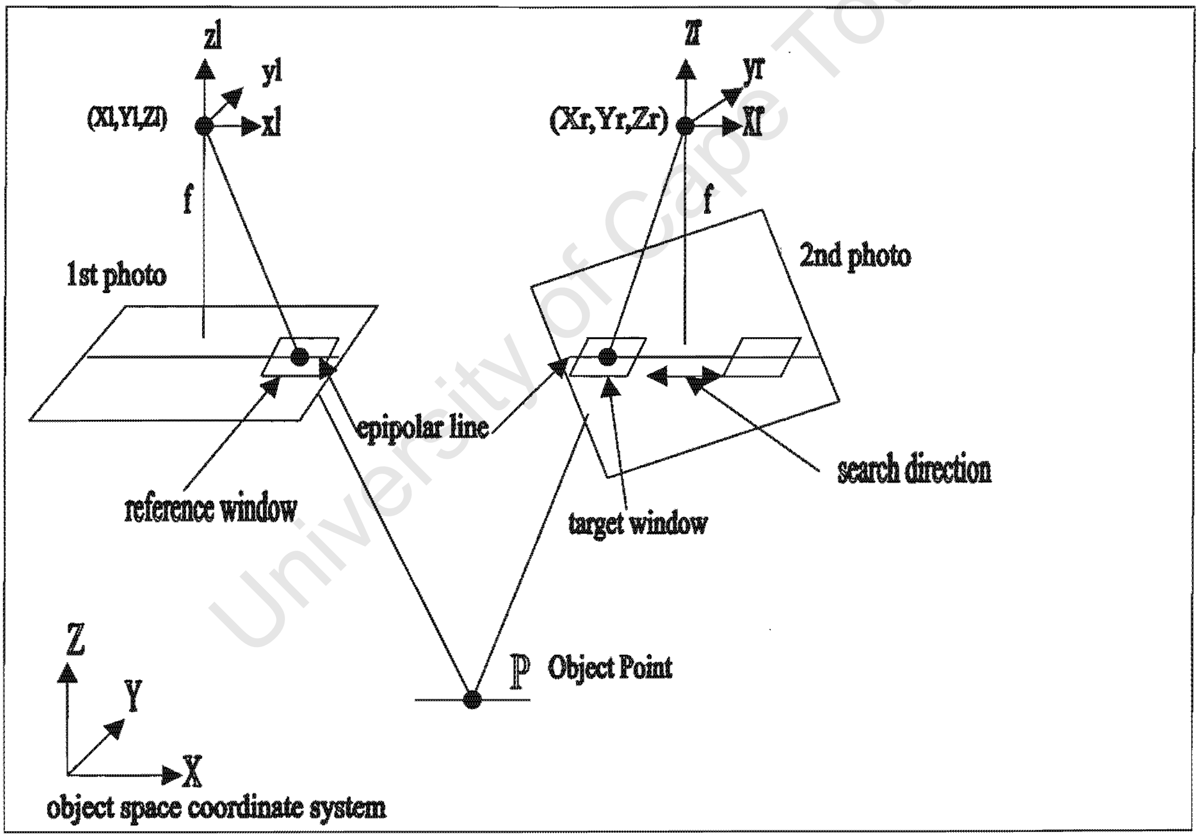


Figure 6. Image correlation scheme (not to scale).

3:1:1:2 SURFACE MODELS BY AUTOMATED IMAGE MATCHING

When interest points are extracted from the overlap area of the reference image, followed by the detection of conjugate points in overlapping target images and the computation of object space coordinates by space intersection, the result is a random point cloud describing the area's surface (Förstner and Gülch; 1987; Canny, 1986; Moravec, 1977; Smit, 1997). Either a regular grid surface model is subsequently obtained by interpolation or a Triangular Irregular Network (TIN) may be created. The density of interest points in most cases tends to be high particularly in areas of high contrast. This requires data thinning by either changing the interest point definition threshold values or by local non-maxima suppression techniques to leave the representative point cloud of the area. Interest points operators normally detect terrain and non-terrain points, and because of this the resulting surface model covers the tops of man-made objects for example, the top of buildings, cars and natural objects such as trees and bushes as well as terrain points.

The image matching process as outlined in § 3:1:1:1 is currently realized as a software module on most digital photogrammetric workstations. Surface models produced automatically on digital photogrammetric workstations usually are not final in themselves in that they have to undergo interactive editing and quality control processes before being put into any application. The surface model for the first study area of this research was generated using the VirtuoZo software package running on a Silicon Graphics Unix based workstation. The surface model for the second study area was generated by an image matching software in-house developed at the Department of Geomatics of the University of Cape Town (Van der Vlugt and Rüther, 1995; Smit, 1997).

3:1:2 AIRBORNE LASER SCANNING TECHNIQUE

Digital surface models can also be generated by air borne laser scanning. Airborne laser scanning generates surface models of a high and homogeneous quality that are ideal for 3-D object reconstruction (Haala and Brenner, 1999; Morgan and Tempfli, 2000). The central component of the system is a laser sensor on board an aircraft which allows direct distance measurements from a point in the aircraft to the topographical terrain surface by runtime measurement of emitted and reflected laser pulses. The object space coordinates of terrain points are then determined by the 3-D polar fixation technique similar to the Tacheometric data acquisition systems found in conventional land surveying. The end product is a range or depth image. Usually the laser beam is emitted perpendicular to the direction of flight, resulting in a strip wise coverage of the terrain surface (Haala and Brenner, 1999; Maas, 1999; Hug, 1997). Additional sensors and components provide the position and orientation of the sensor system during range measurements. Such components include Global Positioning System receivers for the positioning task, and an inertial navigation system for orientation and positioning (OEEPE, 2000; Morgan and Tempfli, 2000).

Laser scanning is capable of positioning terrain points with an accuracy limit of 0.3 m in planimetry, and 0.2 m in height (Maas, 1999). Laser systems provide dense random point measurements along strips which are usually processed to obtain a regular raster. As in the image matching process, laser systems result in an unqualified surface description, that is the object independent distribution of points which results in subsequent considerable computational efforts in their application in object extraction and visualization (Haala and Brenner, 1999; Hug, 1997).

Due to their independence of image texture, laser scanning systems produce surface models of high quality relative to those obtained by the image matching technique. Due to high geometric accuracy, airborne laser-based surface models are more applicable in feature extraction processes than image-matching based surface models which are used only in supporting feature extraction processes, for example in coarse detection of buildings as done in this research (Haala and Brenner, 1997). With regard to feasibility, surface models based on image matching are relatively cheaper than those from laser scanning. Airborne laser scanning is still experimental (OEEPE, 2000) and has not been used in this research.

3:2 DYNAMIC PROGRAMMING OPTIMIZATION TECHNIQUE

3:2:1 CONTEXT

In this research the dynamic programming technique is used to find optimal building contours from approximate ones. Dynamic programming is a method for solving optimization problems. Optimization means finding the best solution among several possible alternatives. Optimization finds application in many fields of science, engineering, industry, economy and others. Many methods of optimization exist, ranging from exhaustive search to Lagrangian variational calculus (Cooper and Steinberg, 1970). Normally, the characteristics of a function to be optimized influence the choice of the method.

Dynamic programming can be used to solve a wide variety of problems. Optimization problems solvable by the dynamic programming technique are those which have variables (not all) in the functional model not interrelated simultaneously (Ballard and Brown, 1992). It is a solution strategy for optimizing problems, which involves sequential decision making. The optimal solution by this technique is obtained through a recursive searching process. A problem solvable by dynamic programming must be convertible into sub-problems, each of which can be expressed by its respective variables. These sub-problems are solved sequentially in such a way that from individual optimal solutions of each sub-problem a general optimal solution of the main problem is achieved (Cooper and Steinberg, 1970). Dynamic programming is applied to each sub-problem to determine the best from many possible solutions. It must be emphasized here that the solution of sub-problems needs to be solved prior to finding the final solution in order to compose solutions of sub-problems to the global solution.

To explain dynamic programming, the following example of finding an optimal path between two given points adapted from Baltsavias (1991) will be used. Assume that it is required to find an optimal path between two given points A and B as shown in Figure 7a below. There are many possible paths between the two points and each path has a cost or a benefit value. To reduce the number of possible paths, let the way between the two points be divided into two stages (also called decision stages) i.e. m1 and m2. Let each stage be with different possible stations i.e. M_i $i = 1,\ldots\ldots N$ (referred to as degrees of freedom).

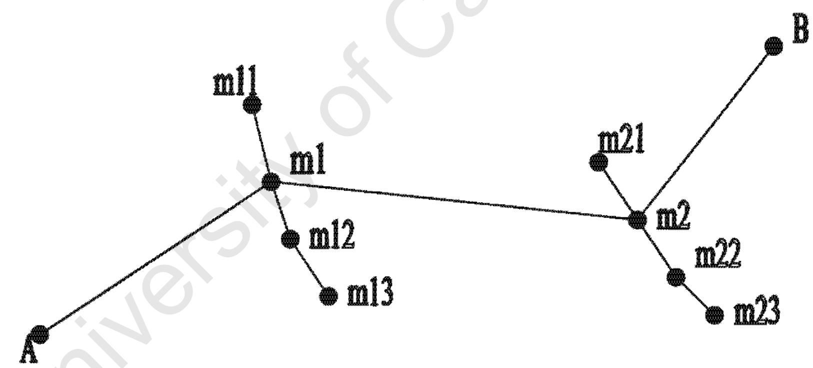


Figure 7a. Example of path nodes in-between two points.

Stations at each stage (i.e. m1 and m2 in Figure 7a) form nodes of possible paths between the two points. For example in Figure 7a, at stage m1, the nodes are m11, m1, m12 and m13 and, at stage m2 the nodes are m21, m2, m22 and m23. Let the paths between any two nodes without any intermediate node be called a primitive path (e.g. path: A-m11 or m12-m21 and so on) and let paths with intermediate nodes (e.g. path: A-m11-m2 or m2-

m21-B and so on) be called partial paths. Further, let the optimal path finding process be carried out sequentially from one stage to the other. At each stage, a decision needs to be made to select an optimal path from the first point i.e. A to that stage. Each primitive path has a cost (or benefit) and the cost of the partial paths is the minimum cost out of the sum of costs of all its possible primitive paths. The cost or benefit of a primitive path is evaluated through a cost or benefit function. Inputs into the cost function are the path's nodes characteristics or attributes (also called node's energy values or forces). The optimal path is selected on the basis of the minimum total cost (or maximum benefit). For a case of M stages and N stations at each stage, there are a total of M*N nodes and at each of these nodes the minimum cost from the first point i.e. A to the node has to be computed. Let the process of computing cost be carried out sequentially through all the stages and nodes. Let

- $c\{(m_{(k,l)}, m_{(i,j)})\}$ be the cost of primitive path between nodes $m_{(k,l)}$ and $m_{(i,j)}$
- $C(m_{(k,l)})$ be the minimum cost of the partial path from the first point to node $m_{(k,l)}$.

Then, the minimum cost at node $m_{(k,l)}$ will be given as:

$$C(m_{(k,l)}) = \text{Min}_{(i,j)} [c\{(m_{(k,l)}),(m_{(i,j)})\} + C(m_{(i,j)})] \qquad (3:3a)$$

Where $1 \leq i \leq k$; $1 \leq j \leq l$; $i+j < k+l$ and $C(m_{(A)}) = 0$.

Equation 3:3a presents the basic principle of the dynamic programming optimization process. In Equation 3:3a at each node $m_{(k,l)}$, the node ($m_{(i,j)}$) which has a minimum cost partial path $C(m_{(k,l)})$ is recorded. Then at the last node, the path that sequentially connects these recorded nodes is the optimal path (This is the back tracking process as explained later in § 5:2:5).

The above example can be formulated to tackle the problem of object extraction from images by defining the optimal object contour as being the one which best satisfies the basic characteristics of an object to be extracted. This means formulating the cost function on the basis of object characteristics. The basic characteristics of an object can be established using the object's generic mathematical model. The cost function can be deduced from the generic object mathematical model. A generic object mathematical model may be defined as formalized statements about object properties and possible instances either verbally or numerically or both. Generic object mathematical models thus describe the manifestation of objects, their possible attributes and their mutual relations. For example, consider a problem of establishing if three given points p_1, p_2, p_3 approximately form an equilateral triangle. A generic mathematical model to describe this condition may verbally be expressed as:

"Included angle between any three points (p_1, p_2, p_3) has to be approximately $60^0 \pm 3^0$"

Mathematically it can be expressed as:

(Bearing p2p3 – Bearing p2p1)-60°<= /3°/

The mathematical model is qualified as being generic because it does not absolutely satisfy given conditions but it only generalizes. For example, in the above case included angles ranging from 57° to 60° are all treated as satisfying the mathematical model.

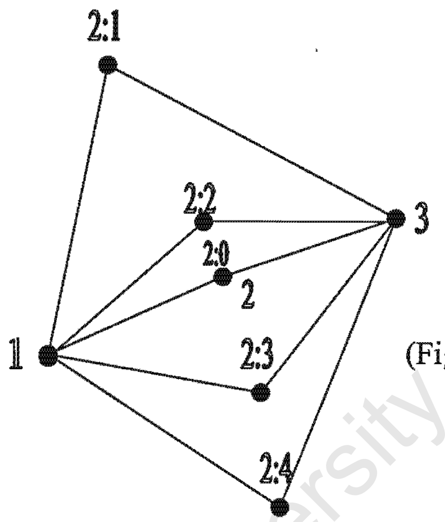
From the dynamic programming example of optimal path finding, as explained above, it can be deduced that characteristics of the dynamic programming optimization technique include:

- the sequential solution of sub-problems,

- the influence of initial sub-problems solutions on the yet to be solved sub-problems in the sequel,
- the possibility of easy imposition of relational constraints on the solution set. For example, in the optimal path finding problem geometric constraints can be imposed on the shape of the optimal path,
- the computational complexity when the number of nodes at decision stages and number of sub-problems (or decision stages) increases and
- the existence of a generic mathematical model from which the cost or benefit of different alternatives can be computed.

It must be noted from the optimal path finding example that prior to the optimization process, energy values at all nodes must be known or computed. Energy at a node may be defined as a value which quantifies a certain attribute or property of a given condition. For example, in the optimal path finding problem, energy at nodes can be computed on the basis of a local criterion, say with respect to bearings from the first point. Apart from local criteria, more global criteria can be used as well. Energy values are used in the cost function to select an optimal path or solution. A cost function is derived from the generic mathematical model and can be a combination of different energy values, each with its own weight.

The concept of energy at a point may be explained by the following example. Let points 1 and 3 in Figure 7b, be fixed and point 2 be free to move to five positions i.e. 2:1, 2:2, 2:3, 2:4 or remain fixed at position 2:0. Out of these 5 possible positions, it is required to select one position which optimally satisfies the collinearity condition between points 1, 2 and 3. By computing the included angle at all possible positions at point 2, and treating it as an energy value, it is possible to select optimal positions for point 2. It can be seen from Figure 7b that position 2:0 will have relatively higher energy value followed by position 2:2 than the rest of the possible positions of point 2. These two positions (i.e. 2:0 and 2:2) will thus be selected as being optimal positions for point 2 under the stipulated collinearity constraint.



(Figure approximately to scale)

Figure 7b. The concept of energy at a point *(positions 2:0 followed by 2:2 have relatively more energy value than positions 2:1, 2:3 and 2:4 with respect to collinearity condition between points 1,2 and 3)*

Finding the optimal position of an object in an image is a typical example of a dynamic programming problem in computer vision. Dynamic programming in image analysis is applicable only if the cost function can be expressed in terms of relationships among neighbouring pixels (Suetens, et al, 1992). Dynamic programming has been applied in tumour delineation in radiographs (Ballard and Brown, 1992). In this case the knowledge that the boundary of a tumour is circular was used to reduce the search space by being the basis for defining the generic mathematical model of the problem. The dynamic programming technique has also been applied in other applications like in semi-automatic road extraction and partial shape recognition in robotics (Gruen and Li, 1997; Gorman et al, 1988). Problems that can be solved by dynamic programming in computer vision can be formulated as variational calculus problems as well. However, the limitations of variational calculus hamper its application in computer vision problems. These limitations are explained by Amini et al (1990) as weaknesses in guaranteeing a global optimal solution, numerical instability, non-convergence and problems in enforcement of hard constraints on the solution. Optimization by dynamic programming is reported by Amini et al (1990) as being a feasible solution compared to variational calculus in image analysis problems. Dynamic programming ensures global optimality in case the optimal solution is among the inspected or visited solutions and it is numerically stable. In addition, relational constraints can easily be enforced on the solution (Bellman and Dreyfus, 1992). Contrary to variational calculus techniques, which require continuous and differentiable mathematical models, dynamic programming can be applied to a discretized form of mathematical models. Many variational problems in computer vision are inherently discrete rendering them ideal for a dynamic programming solution.

3:2:2 MATHEMATICAL MODELS

The dynamic programming technique is formulated in § 3:2:1 to solve the specific problem of finding an optimum path between two points. This section is about the formulation of dynamic programming to solve general problems. Dynamic programming, as already illustrated in the optimal path finding example in § 3:2:1, it is a sequential multistage optimization process that involves the optimization of functions, which are a combination of other sub-functions that are defined by interdependent variables. The main strategy of the dynamic programming paradigm is that a generic mathematical model must exist that relates all variables of the main problem from which a cost function, which reduces the search space, is derived. Dynamic programming optimization is carried out on the basis of finding a solution, which minimizes the cost function. For example to optimize function 'h' (Equation 3:3b) which is defined as a function of sub-functions h_1, h_2 and h_3 which in turn are functions of variables x_1, x_2, x_3 and x_4 i.e.

$$h(.) = h_1(x_1,x_2) + h_2(x_2,x_3) + h_3(x_3,x_4) \tag{3:3b}$$

Where variables x_1, x_2, x_3 and x_4 can take up more than one value i.e. degrees of freedom of the variables. The general procedure is as follows:

1. Find all values of x_1 that optimize function h_1 (using a criterion, as may be stipulated in the cost function of the problem) as well as corresponding x_2 values in an ordered pair set. Record them in the form of a table also referred to as a position matrix, which indicates their ordered pair coordinate values and corresponding energy values.

   i. e $f_1(x_2) = \max_{x1}(h_1(x_1,x_2))$ (Note that functions h_2 and h_3 are not functions of x_1 so they need not be considered at this computational stage)

2. Continue in this manner and optimize variable x_2 in $f_2(x_3)$ i.e.

   $$f_2(x_3) = \max_{x2}\left[f_1(x_2) + h_2(x_2,x_3)\right] \quad \text{then,} \tag{3:4}$$

$$f_3(x_4) = \max_{x3}\left[f_2(x_3) + h_3(x_3, x_4)\right] \qquad \text{and finally,} \tag{3:5}$$

$$\max_{x_i} h = \max_{x4}\left[f_3(x_4)\right]. \tag{3:6}$$

Generalizing the above example to a problem with 'n' number of (h) functions, the following equation is obtained:

$$f_{n-1}(x_n) = \max_{n-1}\left[f_{n-2}(x_{n-1}) + h_{n-1}(x_{n-1}, x_n)\right] \tag{3:7}$$

The final solution x_i which optimizes $h(x_1,x_2,x_3.....x_n)$ is found by back tracking through the position matrices established at each stage of optimization. Back tracking starts from the last optimization stage's position matrix backwards to the first optimization stage's position matrix as illustrated in § 5:2:5.

Computational efforts of dynamic programming optimization technique as noted by Amini et al (1990) increase in the order of nm^2, where:

n – Number of sub-problems or stages of the problem and

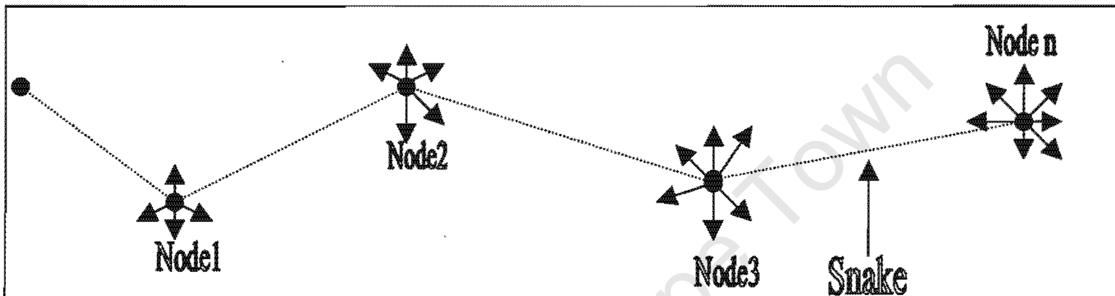
m – Degrees of freedom or directions at each point

If for example each x_i variable in Equation 3:7 has 30 degrees of freedom then for any computation of $f_{n-1}(x_n)$, 30 possible combinations of x_n and x_{n+1} must be computed i.e. 30^2. So the resultant computational effort for n number of sub-problems will be $n\times30^2$ computations. It should be noted that when applying the exhaustive enumeration or total evaluation approach of optimization, there would be 30^n computations of functions h_i. The exhaustive enumeration approach involves computations of all possible combinations of variables in a problem. Thus dynamic programming is far more effective than exhaustive enumeration.

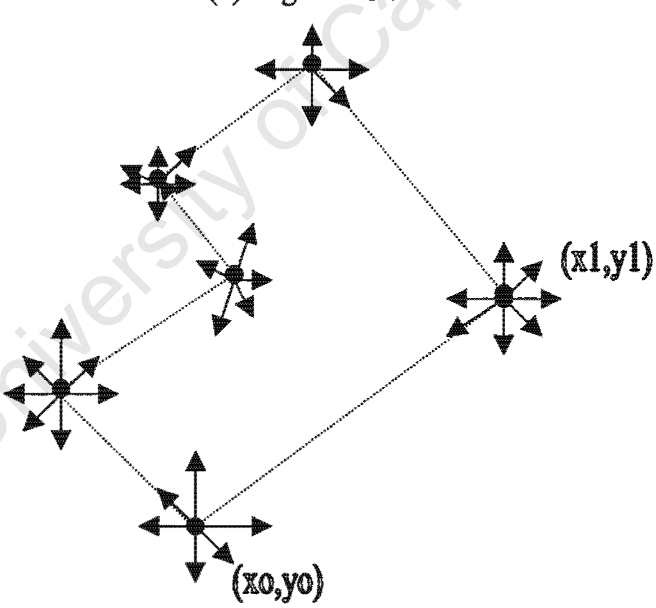
As noted in § 3:2:1, formulation of feature extraction problems in image analysis by the dynamic programming paradigm requires the definition of a generic mathematical model, which relates variables that express the manifestation of a feature in the image. These variables are radiometric and geometric variables because objects in image space are manifested by their geometric and radiometric characteristics. The difficulty in the application of the dynamic programming paradigm is the determination of a generic mathematical model and in some problems the form of this model may be entirely unknown. The adopted generic building mathematical model in this research is as formulated in § 4:9:1.

3:3 ACTIVE CONTOUR MODELS (SNAKES) FORMALISM

In this research, active contour models are used to express approximate building contour positions as being non fixed; that is they can change from one position to another. Solving variational problems in image analysis by dynamic optimization leads to the concept of active contour models. In the optimization process, an edge for example can change from one position to another when different geometric and radiometric constraints are imposed on it. This phenomenon of changing positions when extended to more than one connected edge leads to the concept of snakes or deformable snakes or active contour models (terms used synonymously). A snake may also be described as an elastic curve, defined by its nodes and corresponding connecting lines (see Figure 7c). In Figure 7c two types of snakes are shown, the general snake (Figure 7c(a)) and rectangular constrained snake (Figure 7c(b)). In rectangular constrained snakes, nodes can change position but in such a way that they will always satisfy the constraint. General snakes have been used in semi automatic feature extraction in many applications (Gülch, 1990a, and 1990b; Gruen and Li, 1997; Amini et al,1990). The main strategy of the snake's formalism in feature extraction is working from the approximate contour position of a feature (which can change positions) and finding the best contour position by dynamic programming optimization. It must be noted that only nodes of the snake are free to move (have degrees of freedom) thereby allowing the snake to slither. By imposition of geometric and radiometric constraints on the movement of the snake, it is possible to control the movement such that nodes are attracted to designated feature corner points. The final position of the snake is the one which optimally meets the object characteristics as stipulated in the generic mathematical model. Active contour models offer a link between high and low level image interpretation processes with the easy possibility of introducing some degree of automation (Gülch, 1990a, and 1990b; Amini et al,1990). In delineating an object in a digital image by the snakes approach, the snake expands and contracts to points of various attributes thereby bridging over photometrically weak boundaries.



(a) A general Snake



(b) A rectangular constrained snake (not applied in this research).

Figure 7c. Examples of 2-D snakes.

NOTE:

- (a) General Snake (Arrows indicate the degrees of freedom or directions the node can move to)
- (b) Rectangular constrained Snake (Nodes can move to indicated directions BUT they should always satisfy the rectangularity constraint) NB: Rectangular constrained snakes were not applied in this work.

A snake from its initial or approximate position slithers to delineate an object in the image space. The approximate position of a snake can be provided either automatically or manually by a human operator interactively defining its initial position. This needs to be as close as possible to the object to be extracted (Gruen and Li, 1997). In this research, general snakes are used in the delineation of buildings. Approximate building contour positions are obtained automatically by processing using the technique of region growing constrained by edges. Region growing is carried out from elevation blobs centre points. Optimal building delineation is achieved by approximate contour position optimization using the dynamic programming technique (Gülch, 1990b).

The behavior of a snake in the image space is controlled by internal and external energies or forces acting on it. The total energy of the snake is the sum of all energies acting on it. This depends on where the snake is placed and how its shape changes locally in space (refer § 3:2:1 for definition of energy). A snake is attracted to features of interest by optimizing i.e. minimizing or maximizing the snake's total energy. During the optimization process the snake is deformed to find an optimal position by considering all constraints that are imposed on it by internal and external forces. Forces acting on a snake are radiometric or image related forces (external) and object geometrical forces (internal). Internal forces offer the possibility of introducing geometric constraints on the shape of the snake, whereas external forces attract the snake to the features of interest in the image e.g. to brighter pixels, to strong corner points, to strong edge magnitude points and so on.

Along-side the recipe of optimization for feature extraction, Pavlidis and Liow (1990), approached the problem of contour optimization by the application of a merit function. Apart from the formulation of energy values, their approach is analogous to the snakes approach. The merit function applied took into consideration contrast and smoothness (curvature) to minimize artifacts resulting from region growing and the edge detection processes and was expressed as:

$$g(w) = \frac{1}{L}\sum_i \left[\left|\nabla I(w(t_i))\right| - \alpha.\kappa(w(t_i)) - \gamma\left|\phi'(w(t_i))\right| \right] \tag{3:17}$$

Where:

$w(t_i)$ = (x(t_i),y(t_i)) is a contour image point,

$|\nabla I(w(t_i))|$ is the magnitude of the image gradient at point $w(t_i)$,

κ ($w(t_i)$) - Curvature at point $w(t_i)$,

ϕ' - First order derivative of the phase part of the image gradient at point $w(t_i)$,

γ and $\propto$ - are a scale factors.

The approach used by Pavlidis and Liow (1990) is ideal for linear features like roads. The application of this approach in building extraction necessitates modifications of the merit function so as to take into consideration building properties like rectangularity, sharp building corners, and so on.

The snakes and dynamic programming techniques are used in this research to extract buildings in informal settlement settings. A standard C code is developed and implemented as explained in detail in Chapters 4 and 5.

3:4 GRAY SCALE MATHEMATICAL MORPHOLOGICAL FILTERING.

Gray scale mathematical morphology is used in this research to accentuate height discontinuities in-between very close buildings as obtained by the image matching technique. This is done by treating heights as gray values. Mathematical morphological operations result in altering input data but preserving essential data characteristics and eliminating irrelevancies. In image analysis mathematical morphology operations preserve object shape characteristics. The shape of an object in a digital image can easily be perceived when the image is in binary form. A binary image is a processed gray scale image in which pixels with value of 1 constitute object pixels and 0 the background or vice versa. Binary images are preferred to gray scale images in the extraction of object shapes because they can be manipulated easily. Thus mathematical morphology operations in image analysis are commonly applied to binary images and they are referred to as binary mathematical morphological operations. Binary mathematical morphology operations are used to remove object shape artifacts in image analysis. The main operations are binary dilation and erosion.

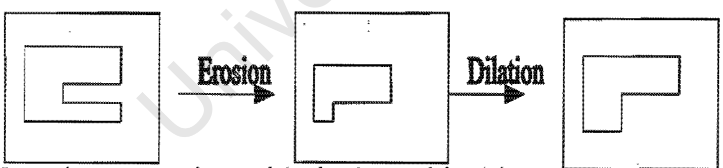
Binary dilation is used to fill small holes and narrow gaps in objects or expand image objects, whereas binary erosion shrinks the image objects. Binary dilation and erosion are defined by Gorte (1999) as binary image transformations with a binary structuring element or kernel (terms used synonymously) in which the operation performed is of a correlation type and not a convolution. The following predicates are the basis of this definition:

- Binary dilation – If all designated pixels (with the value of 1 (one) in the structuring element) correspond to 0 (zeros) in the resulting sub-image THEN the central pixel of the sub-image is assigned the value of 0 (zero) ELSE it is assigned the value of 1 (one).
- Binary erosion - If all designated pixels (with the value of 1 (one) in the structuring element) corresponds to 1 (one) in the resulting sub-image THEN the central pixel of the sub-image is assigned the value of 1 (one) ELSE it is assigned the value of 0 (zero).

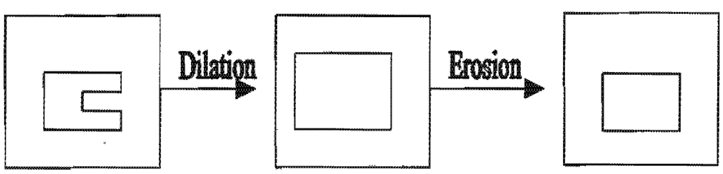
The design of the structuring element is critical in mathematical morphology operations as it greatly affects the results. Binary opening and closing operations are derivatives of binary dilation and erosion, such that when a dilation operation is followed by an erosion operation it leads to closing and the reverse leads to opening (refer Figure 7d).

1	1	1
1	1	1
1	1	1

(a) A 3 by 3 binary kernel



(b) Opening operation with the kernel in (a).



(c) Closing operation with the kernel in (a).

Figure 7d. Example of opening and closing of binary figures. (NOTE: The effect of change in shape is by one pixel but in the above diagrams the effect is exaggerated to make it more noticeable).

Binary opening is used to eliminate specific artifacts that are smaller than the structuring element; while binary closing connects objects that are close to each other, fills up small holes and smoothes the object by filling up narrow gaps (Saradjian and Amini, 2000). Binary morphological operators are currently realized on most commercial image processing software packages.

The binary morphological operations i.e. dilation, erosion, opening and closing can be extended to gray scale images (Haralick, et al 1987). In gray scale images, erosion and dilation are achieved by taking the minimum and maximum gray values within neighborhoods covered by the kernel. Let F and T be the domain of the gray scale image 'f' and the gray scale structuring element 't' respectively. The gray scale dilation and erosion of 'f' with 't' denoted by $(f\oplus t)(x,y)$ and $(f\ominus t)(x,y)$ respectively are defined as:

- $(f\oplus t)(x,y) = \max_{(x-m,y-n)\in F(m,n)\in T}\{(f(x-m,y-n) + t(m,n)\}$ (3:17a)
- $(f\ominus t)(x,y) = \min_{(x+m,y+n)\in F(m,n)\in T}\{(f(x+m,y+n) - t(m,n)\}$ (3:17b)

Where (m, n) – defines the size of the kernel (Saradjian and Amini, 2000).

A detailed description of gray scale erosion and dilation is given by Haralick et al (1987), Haralick and Shapiro (1994) and Saradjian and Amini (2000).

The gray scale mathematical morphological operations are reported to be applicable in image contrast enhancement and in the computation of an approximate topographic earth surface model for the purpose of building extraction (Haralick et al 1987; Weidner and Förstner, 1995; Hug, 1997). Gray scale morphological operations are yet to be realized in most commercial software packages, but this may be attributed to their limited applications.

3:5 DIGITAL ORTHOPHOTOGRAPHY, REGION GROWING AND EDGE DETECTION.

3:5:1 DIGITAL ORTHOPHOTOGRAPHY

In this research, building extraction is carried out from orthoimages. An orthoimage or orthophoto is a georeferenced image created from perspective photographs or other remotely sensed imagery in which the displacement within the image, due to the sensor orientation i.e. tilts, relief displacements as well as image distortions due to sensor, has been removed. An orthoimage is an orthogonal projection and thus can be used as a map or as a backdrop for a GIS, screen digitizing, map revision 3-D visualization and so on. Orthoimages are used in many countries to support classical mapping tasks. Black and white and coloured infrared orthoimages are favored today in various applications especially in mapping, planning and environmental disciplines. Orthoimages are advantageous because of their pictorial qualities for interpretation and in the richness of the information content (Grenzdörffer and Bill, 1994). Digital orthoimages are also advantageous compared to their analogue counterparts especially with respect to flexibility, short production time and low cost, production of derived products, easy manipulation in radiometry and combination with other data sets (Grenzdörffer and Bill, 1994; Baltsavias, 1996; Monier et al, 1999; Madani, 2000).

Inputs to the orthoimage production process are digital images (analogue images have to be scanned with high resolution); interior and exterior orientation parameters of the images and a digital terrain model. There are two methods of digital orthoimage production and these are the direct and the indirect methods. Inputs to the direct method are image coordinates of pixels to be ortho-rectified whereas inputs in the indirect method are object space coordinates of points to be ortho-rectified. The direct method involves computation of object space coordinates of ortho-rectified pixels using interior and exterior parameters (i.e. space intersection) followed by re-sampling (in the object space) to establish a regular grid of pixels. The indirect method involves computation of ortho-rectified image coordinates of pixels by space resection followed by gray level re-sampling (in ortho-image space) to determine pixels gray level values. Orthoimages used in this research are generated by the direct method. In the orthoimage process, single scanned aerial photographs or satellite imagery are digitally rectified to an orthogonal projection by processing each image pixel individually. The entire process is illustrated in Figure 8 and can be automated on Digital Photogrammetric Work Stations. The quality of orthoimages depends mainly on the accuracy of the digital terrain model used and on such other factors as differential geometric rectification techniques, pixel size, resampling and mosaicking methods.

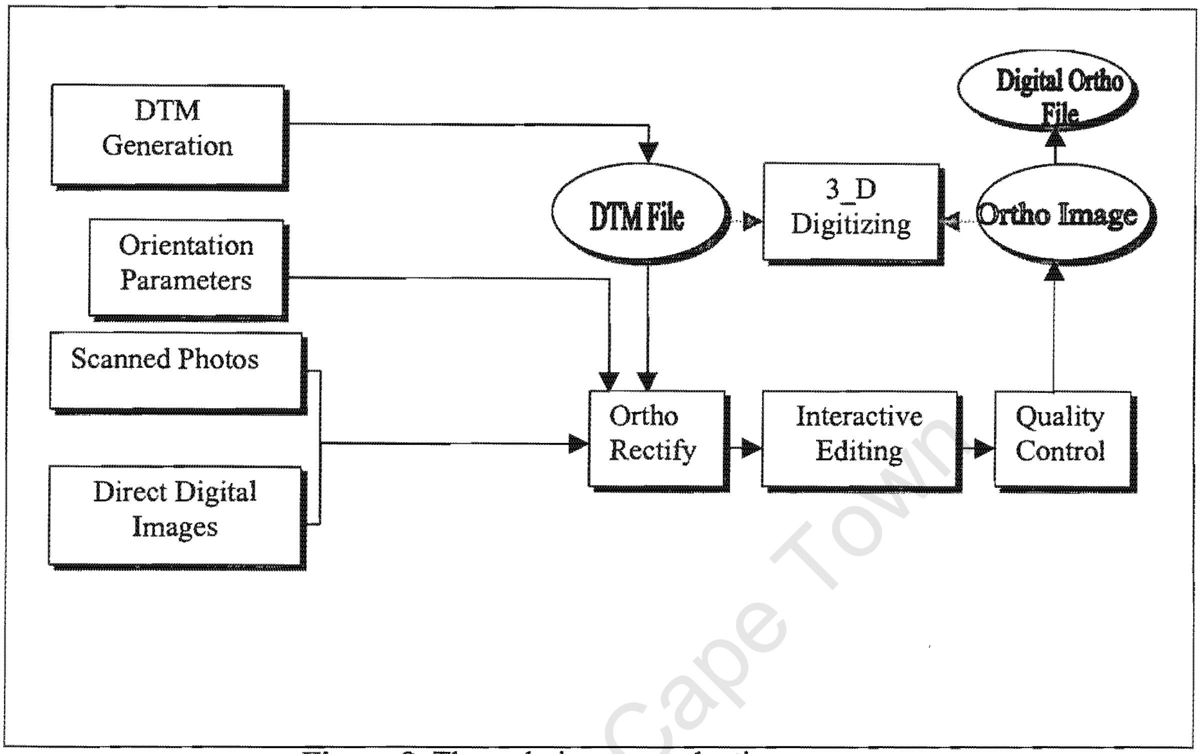


Figure 8. The orthoimage production process.

There are wide application areas for orthoimages ranging from feature extraction, change detection, GIS analyses, generation of image maps and the improvement of multispectral classification to visualization (Baltsavias, 1996). It is accurate and reliable to do 2-D measurements directly on orthoimages and interpreting heights online from a digital terrain model, as shown in Figure 8 above. In this way 3-D digitization can easily be performed for all terrain features, though non-terrain objects cannot be extracted very accurately. Feature extraction in orthoimages is ideal and convenient particularly to novice users because of the non-interpreted imagery, higher similarity with the landscape and enhanced image quality compared to conventional line maps (Grenzdörffer and Bill, 1994). The ability of orthoimages in feature extraction extends to change detection where multispectral orthoimages can be used to highlight regions of possible change and thereby guide a manual or semi-automatic change extraction scheme. Other advantages to using orthoimages in feature extraction are that expensive stereo viewing capabilities are not needed and training in stereo measurements is therefore unnecessary. Orthoimages are thus ideal for use in informal settlements areas were the levels of literacy and technological advancement is low.

Orthoimages are handy photogrammetric products, which constitute a sufficiently accurate geometric reference. In this research, buildings are extracted from orthoimages under the assumption that geometric displacement is not affected by building elevation as buildings are not very tall. Most buildings in the study areas have heights in the range of 3-4 m above the average ground.

3:5:2 REGION GROWING

Region growing and edge detection techniques are applied in this research to define approximate building contours. Region growing is a method of segmenting an image into homogeneous areas with respect to certain uniformity criteria. The criterion can be on the basis of texture, brightness, color, variances and other factors. Segmented regions need to be simple and without inside holes. Adjacent regions of segmentation need to have significantly different values with respect to the uniformity criterion. Boundaries of each segment should be simple, not ragged and must be spatially accurate (Haralick and Shapiro, 1985; Tanimoto, 1987). In the real world, most man-made objects are manifested as homogeneous regions in digital images. This is because man-made objects, like buildings, are characterized by being made up of almost the same building materials. It is therefore logical to approach the problem of extraction of man-made objects on the basis of some form of homogeneity. However, due to noise, occlusions and different illumination conditions, segmentation of man-made objects is usually accompanied by artifacts which need to be corrected before the results are put into high level or semantic image interpretation processes (Koch, and Kashyap, 1987; Zongjian et al, 1990; Pavlidis and Liows, 1990).

In general, there is no universal theory of image segmentation (Haralick and Shapiro, 1985; Fua, 1982). However, many segmentation techniques exist in the literature. Most of the techniques are ad hoc and some are designed for a specific class of images.

Image segmentation techniques can be classified on the basis of the following schemes:

- Measurement Space Guided Spatial Clustering:

This approach is the segmentation of an image on the basis of measurement space clusters. A measurement space is the domain in which pixels are measured, for example on the basis of pixel intensity values, texture and so on. In this method the entire image is read into memory, converted into respective measurement space and on the basis of variations in the measurement space, it is grouped into clusters or segments. These clusters are used to assign, the most probable cluster or label to each pixel in the image. The resulting image segments are connected components of the pixels having the same label. The accuracy of this technique depends on how well the objects of interest in the image separate into distinct measurement space clusters. The commonly applied clustering approach is the pixel intensity measurement space (using histograms), in which it is assumed that homogeneous objects in the image manifest themselves as peaks in image histograms.

- Single Linkage Region Growing:

In this technique each pixel in an image is regarded as a central pixel of an hypothetical region and neighboring pixels whose properties are similar enough, are declared as belonging to the same region. The easiest single linkage scheme defines " similar enough" by pixel differences i.e. two neighboring pixels are similar enough if the absolute value of the difference between their gray value is below a predefined threshold value.

- Hybrid Linkage Region Growing :

The technique assigns a property vector to each pixel in an image. The property vector depends on a neighboring window which is defined at each pixel. In this method, pixels are declared similar on the basis of similarity of their neighborhoods. Similarity is defined as a function of neighboring pixels, and this makes the technique perform well in noisy data. This is because noisy pixels are assimilated into respective neighbouring windows and thus their effect does not show up in the result.

- Centroid Linkage Region Growing :

This approach involves scanning an image in some predetermined manner such as left-right or top-bottom to define initial image segments mainly on the basis of pixel intensity variations. Scanning is carried out only on the part of the image and initial segments defined are representative of the entire image. The intensity value of each pixel in the rest of the image is compared with means of the initial segments. If the pixel intensity value is close enough (user defined) to the mean of any initial segment then the pixel is added to that segment and the mean of the segment is updated.

- Spatial Clustering:

This technique is based on segmenting the image by simultaneously combining the measurement space guided spatial technique with ordinary spatial region growing. It is a combination of the histogram mode seeking clustering approach with other region growing techniques, mostly the single linkage technique. Segmentation is performed by locating all peak pixels in the measurement space histogram, followed by single linkage region growing from each peak pixel's image position.

- Split and Merge:

This technique starts by assuming the entire image to be an initial segment. Then successively it splits each current segment into quarters as long as the segment is not homogeneous enough, on the basis of texture, variances or other criteria. The process continues recursively until all resulting segments pass the homogeneity test (Haralick and Shapiro, 1985; Tanimoto, 1987).

Some of these segmentation schemes are realized in commercial image processing software packages. In this research, region growing using the centroid linkage method is applied in defining approximate building regions. In the application of the region growing method using the centroid linkage method, building regions are established by region growing from elevation blob's centre points. The elevation blobs used are those which previously are hypothesized as containing building roofs as from a digital surface model.

3:5:3 EDGE DETECTION

An edge in an image refers to places in the image where brightness values jump or change abruptly. A digital edge is the boundary between two pixels that appears when their brightness values are significantly different. Edge detection is a valuable aid in image interpretation and most of the information in an image is contained in edges. Experiments with the human vision system show that edges in an image are extremely important, and objects can be recognized from only their crude edge outlines (Pratts, 1991). The edge detection process therefore is segmentation of an image based on the detection of brightness discontinuities. Many algorithms exist for detecting edges in the literature. To make sense of out of detected edges, a process of edge interpretation and analysis is thus necessary.

A feasible approach to edge detection and interpretation is to first transform the image into an intermediate image of local gray level discontinuities and compose these later into meaningful edges. The principle of edge detection is that an image array is a result of sampling of a real-valued function *(f)* defined over the domain of the image and that function *(f)* is continuous. The images are discrete 2-D approximations of such a function. From this point of view, the jumps or discontinuities in the values of *(f)* must obviously refer to points of high first order derivative of *(f)*.

Other approaches of edge detection are surface fitting which is relatively computationally involved and application of edge operators. An edge operator is a mathematical process (or its computational equivalent) operating within a small spatial extent designed to detect the presence of a local edge in an image. In this approach, pixel differences are computed along orthogonal and/or oblique directions at all points in the image and the decision as to whether there is an edge or not is made on the basis of an edge gradient magnitude. Commonly applied 3 by 3 gradient operators include Prewitt's, Sobel, Roberts, Frei-chen and others (Mather, 1987; Haralick and Shapiro, 1992; Pratts, 1991; Ballard and Brown, 1992; Richards, 1993). In this research, the Sobel edge detection approach is used in defining approximate building contours as described in Chapters 4 and 5.

CHAPTER 4

BUILDING EXTRACTION DESIGN STRATEGY

4:1 CONTEXT

This chapter is about the flow of activities and the detailed description of procedures used in the design of the proposed building extraction method, along with some initial implementations. There exists two control strategies of automatic image interpretation, which are the top-down and the bottom-up control strategies. Top-down control strategies (also called model-based) are goal directed processing techniques which involve hypothesis generation on the basis of the contextual knowledge of objects, the identification of the focus of attention areas in the image space and object extraction and reconstruction. On the other hand, bottom-up control strategies use image data to generate object primitives usually by application of local statistical based operators (low level image interpretation) followed by mid and high-level image interpretation processes. Mid-level image interpretation processes involve the aggregation of image primitives or tokens to generate collateral object features, which are used by high-level (semantic) image interpretation processes which are object recognition, extraction and reconstruction.

The building extraction system proposed and tested in this research is based on a top-down control strategy. The system starts by object's hypotheses generation, verification, projection of hypotheses into the image domain and finally by objects delineation directly from image pixels (refer Figure 9). The adopted control strategy is contrary to conventional photogrammetric feature extraction processes which are mainly data driven (bottom-up) but it is commonly applied in computer vision applications as in industrial inspections, navigation, robotics and others.

The basic principles of the proposed building extraction strategy are detection and extraction of buildings in orthoimages. It is difficult hypothesizing objects in images by relying merely on their gray values. Fusion of other information sources (cues) is necessary in this process (Shufelt and McKeown, 1993; Li, Mason and Rüther, 1998). This is because gray values are a result of a combination of many factors such as surface property, illumination conditions, atmospheric conditions just to mention a few. It is therefore not possible to differentiate a contribution from each of these factors for object detection and recognition purposes. In addition to that, the depth information is usually collapsed in the imaging process and as a result images contain reduced information.

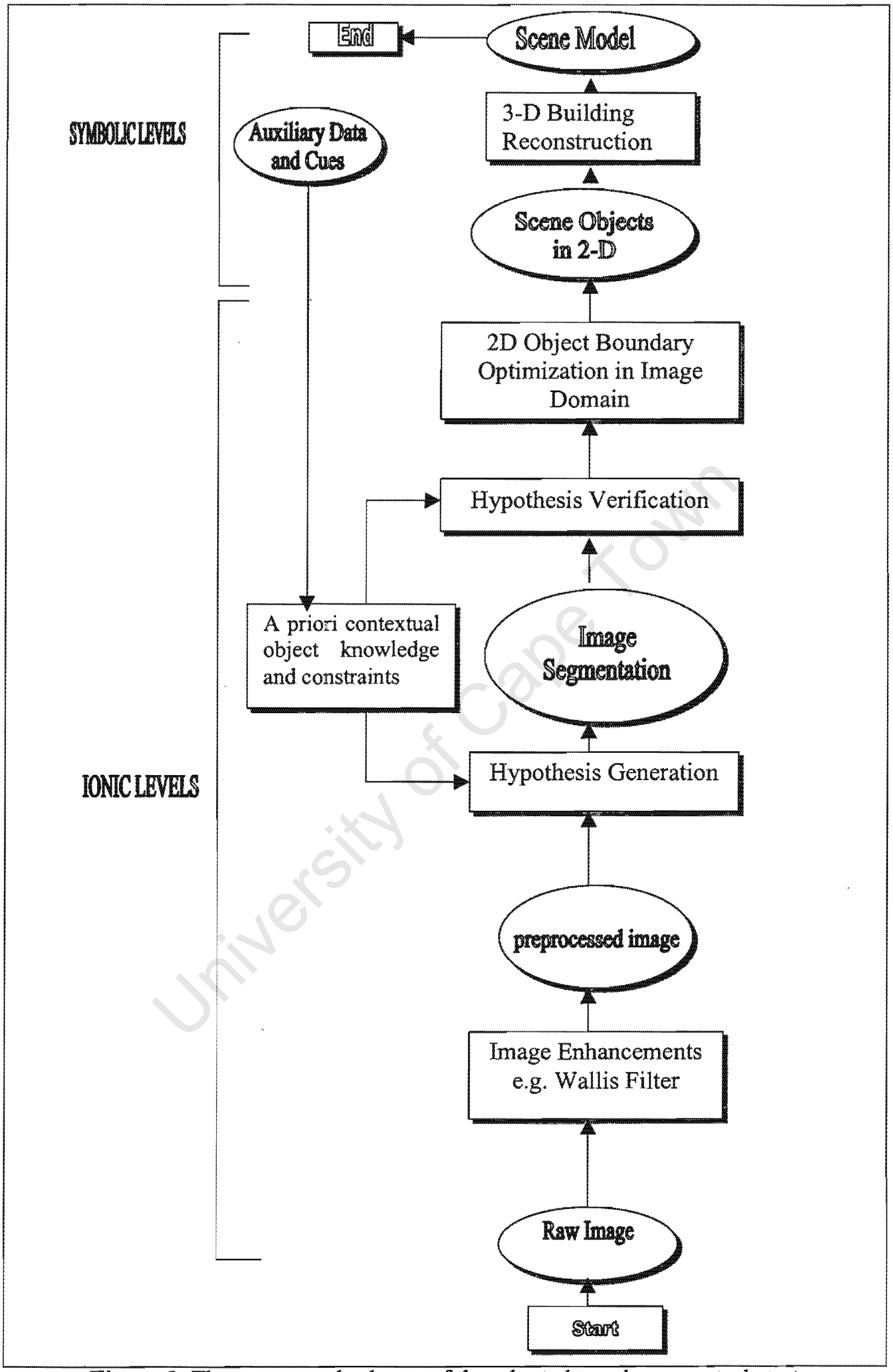


Figure 9. The conceptual scheme of the adopted top-down control strategy.

The application of additional or auxiliary data such as size, position of objects and so on, facilitates the recovery of information from images. In general, image analysis requires a priori domain knowledge or cues to reduce the search space. Domain knowledge is problem-oriented knowledge, which is relevant to a particular application. A priori domain knowledge used in this research is derivered from digital surface models. Surface models model non-terrain objects like buildings as well as the terrain. This characteristic makes digital surface models ideal for building detection (Weidner and Förstner, 1995; Weidner, 1997; Baltsavias et al, 1995, Rüther et al, 1997; Mason and Baltsavias, 1997; Haala and Brenner, 1997; Haala and Hahn, 1995; Hug, 1997; Mass, 1999; Paparoditis et al, 1998). As already mentioned in Chapter 2, the use of digital surface models in building extraction is attributed to the fact that a DSM inherently provides a geometric description of a scene derived from aerial imagery or airborne laser scanner data (Brunn and Weidner, 1998).

The proposed building extraction system's flow of activities is as shown in Figure 10. The major steps in the proposed building extraction method include:

- image preprocessing,
- generation of a digital surface model,
- generation of an approximate digital terrain model,
- generation of orthoimages,
- altimetric thresholding of surface models creating raised structure blobs,
- separation of merged blobs,
- building hypothesis verification,
- generation of approximate building contours by region growing controlled by edges,

- formulation of approximate building contours into active contour models (snakes) and optimization of approximate building contours by dynamic programming. i.e. 2-D building extraction and
- 3-D building reconstruction.

Details of these steps are explained in the remainder of this chapter.

The procedure described in this thesis is not fully automatic because most processes are designed to be interactive, with the operator guiding the system in making crucial decisions at various stages. Nevertheless, it is felt that this procedure is a major step towards the extraction of buildings in informal settlements and that it adds significantly to the state of the art.

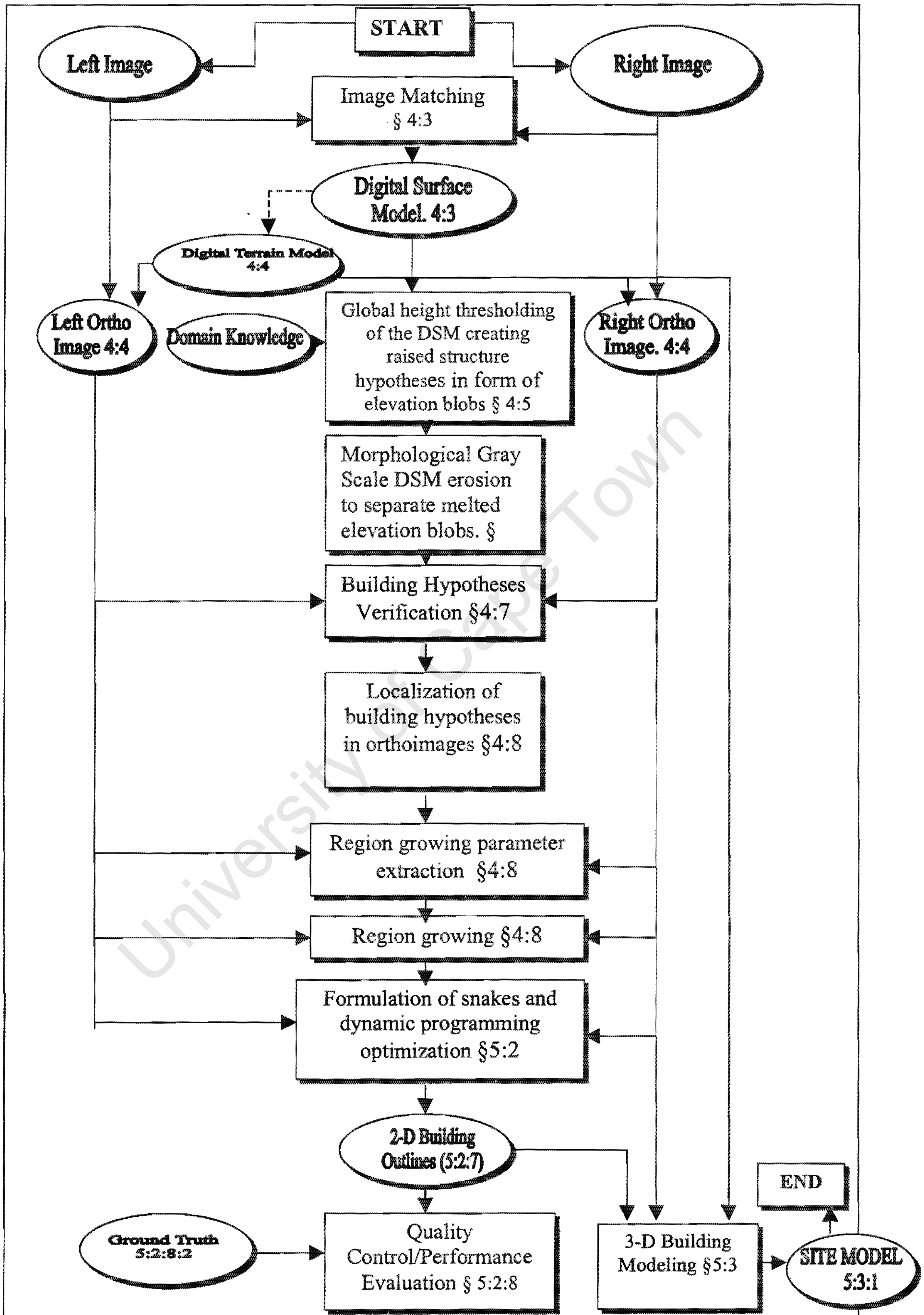


Figure 10. Flow of activities of the proposed building extraction system. (number in boxes indicate the implementation section in the thesis).

4:2 IMAGE PRE-PROCESSING

Building extraction started by image preprocessing which involved image enhancements aimed at improving the quality of the images in terms of contrast in preparation for subsequent feature extraction processes. The Wallis Filter was applied to the image of Marconi Beam area, which was acquired by the small format digital camera Kodak DCS 460, and the results are as shown in Figure 11b. The image data set for the Manzese area was from scanned aerial images. The images had satisfactory contrast that required no enhancements (see Figure 11c).

The Wallis Filter was applied because it is characterized by locally stretching the gray value range to fit certain target values (Mason and Baltsavias, 1997). Wallis filtering is a useful tool in image enhancement, and is more effective in images with big global contrast and whose gray values stretch over a small range (Gruen and Li, 1997). The Results of the image pre-processing stage are as shown in Figure 11.

(a) Original Marconi image

(b) Wallis Filtering Marconi Image

(c_1) Site1

(c_2) Site2

(c) Image quality - Manzese Area (not Wallis filtered)

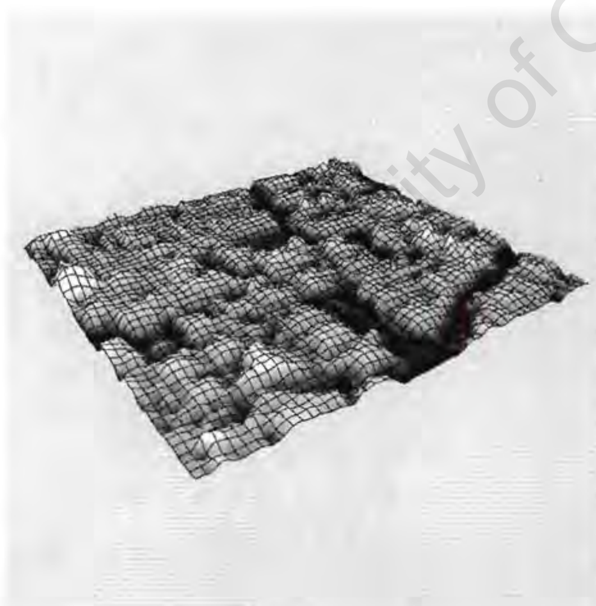
Figure 11. Results of image preprocessing and quality of images used.

4:3 GENERATION OF DIGITAL SURFACE MODELS

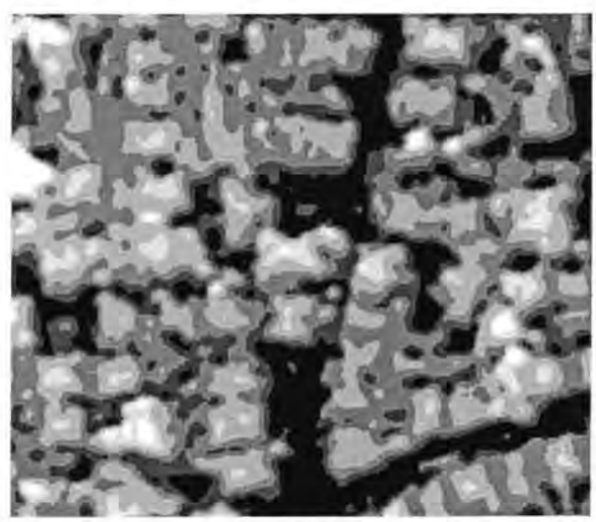
The Digital surface models used in this research have been generated by the area based image matching technique (refer § 3:1:1). This is the current, and adequate technology, though it is acknowledged that more accurate digital surface models can be obtained using the laser scanning technique (Haala and Bremmer, 1997).

The surface model for the Manzese site was generated using VirtuoZo 3.0 software running on a Unix based Silicon Graphics Digital Photogrammetric Workstation. For the Marconi Beam area, the surface model was generated by an image matching software based on the least squares matching (MPGC) which was developed in-house at the Department of Geomatics, University of Cape Town (Smit and Rüther, 1996). Two software packages were used because the data sets had slightly different characteristics particularly with respect to format and lens distortions. Whereas VirtuoZo is designed to process conventional (metric) aerial photographs, the in-house software was designed to process mainly non-conventional photographs.

Inputs to both image matching software packages were camera calibration data, ground control points, flying height and high resolution overlapping digital images. The spatial resolution of input images is critical to the accuracy of the generated surface model. In view of this, images of the Manzese site were scanned at 15 μm, which at 1:12,500 photo scale, is equivalent to a ground resolution of 20cm, whereas for the Marconi Beam area it was 25cm. A dense cloud of random points was matched generating a random DSM from which a regular grid DSM was obtained by interpolation. It must be noted that interpolation was done without regard to the type of features expected to be found on the ground. For informal settlement building extraction purposes, it was necessary to create a dense DSM, that is, the one with small grid spacing so as to adequately capture height discontinuities in-between very close buildings. The use of coarse grid spacing may lead to buildings being poorly modeled or not modeled at all. In the view of this, surface models at a regular grid spacing of 0.20m and 0.25m for the Manzese and Marconi Beam areas respectively were generated and they are shown in Figures 12 and 13.

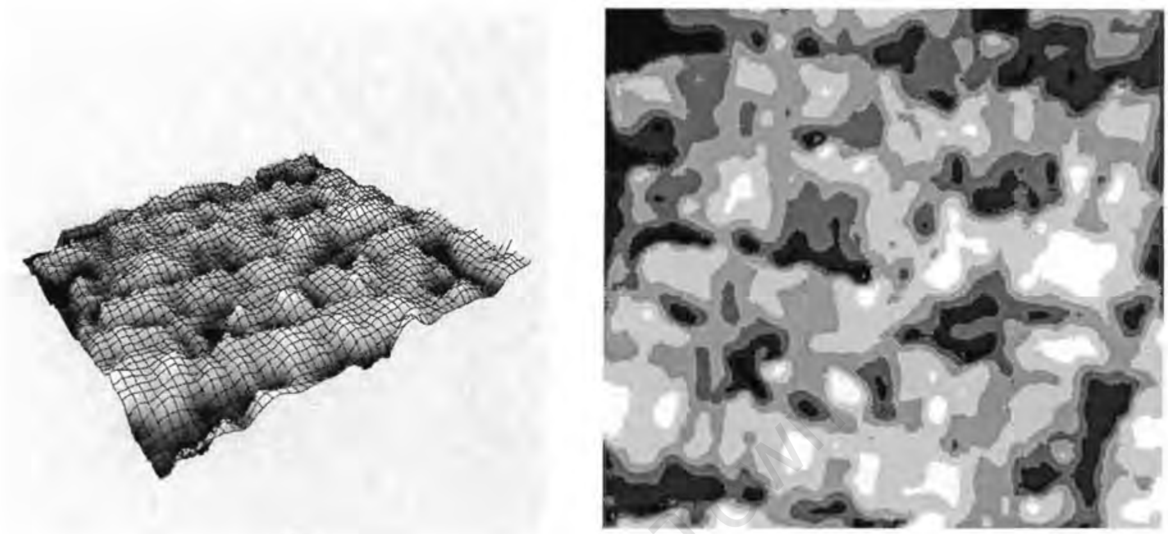


(i)



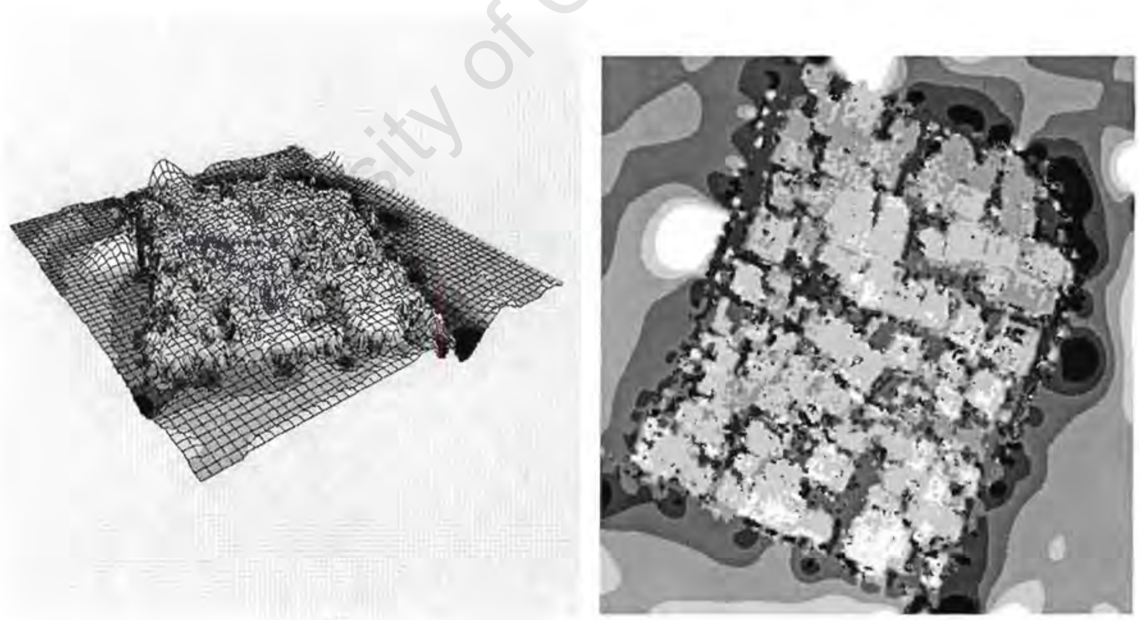
(ii)

(a) Manzese Site 1



(i) (ii)

(b) Manzese Site 2



(i) (ii)

(c) Marconi Beam

Figure 12. Generated surface models (i) Wire Mesh surface model draped on image

(ii) Gray coded surface model.

4:4 GENERATION OF ORTHOIMAGES

Orthoimages of the study areas were produced after the generation of digital terrain models. This was already in existence for the Marconi Beam site and it was automatically generated using the VirtuoZo software package for the Manzese site. Interactive editing was applied to the digital surface models prior to digital terrain model generation. Digital surface model editing was necessary to eliminate blunders and gross errors. It took approximately four hours to edit the few blunders that were present in the digital surface model.

For the Manzese site, surface model editing was carried out interactively on screen by superimposing the surface model (in form of contours) on a stereo model. Contours that did not appear to match with the corresponding features from the stereo model were identified as being erroneous and were subsequently modified or deleted. Viewing was achieved by the use of a polarized glass system. After surface model editing, a digital terrain model was produced and subsequently an orthoimage at a pixel size of 0.06 m was generated (see Figure 13a). For the Macorni Beam site, orthoimage generation at a pixel size of 0.2 m was carried out on the Helava Socet Set Digital Photogrammetric Workstation (see Figure 13b).

(a) Manzese Site

(b) Marconi Beam Site

Figure 13. Generated orthoimages.

4:5 THRESHOLDING OF DIGITAL SURFACE MODELS

Building extraction from surface models is normally based on altimetric region thresholding (Förstner and Weidner, 1995; Mason and Baltsavias, 1997). Points on the surface model are points which are visible from the sensor. A surface model point can thus be either a terrain point or a non-terrain point (i.e. on a raised structure). Due to the fact that building points along walls or facades are not visible from sensors, surface model points which represent building points generally must be points of a roof structure. The building extraction process from surface models thus turns out to be a roof recognition scheme. In principle, roof points can easily be extracted by altimetric region segmentation of the surface model. In view of that, digital surface models are used in the detection of building roof areas from which a detailed building extraction process can be initiated. Detected building roof areas are projected into the orthoimage where they define focus of attention areas. Detailed building delineation is subsequently carried out using orthoimage data.

A common approach to DSM's altimetric region extraction is through the creation of a "normalized" surface model (Förstner and Weidner, 1995; Brunn and Weidner, 1998; Mason and Baltsavias, 1997; Hug, 1997). A "normalized" surface model is obtained by differencing a previously established digital terrain model (DTM) from the DSM. The DTM describes the earth's topographical surface, therefore it is used in providing information about building heights approximately referenced to the terrain surface. If the DTM does not exist it may be produced separately by either photogrammetry or it may be approximated through techniques such as the gray scale mathematical morphological operations (Förstner and Weidner, 1995; Brunn and Weidner, 1998). It is rare to find an existing DTM in developing countries therefore the adoption of the latter approach may be the only alternative. However, the effectiveness of mathematical morphology depends on the design of a kernel. For example, Förstner and Weidner (1995) applied an approach which used a kernel with constant height values and satisfactory results were reported. In their approach, the morphologic gray scale technique was applied to erode the surface model until the approximation of the topographic surface converged towards check points in non-building regions and interpolated well within the building regions. Check points are points (preferably in open areas) with known or easily determinable height values. Where buildings are very close to each other, as in informal settlement areas, check points are difficult to obtain. The approach of using normalized digital surface models in building extraction is subjective (particularly with the choice of a kernel) and its contribution to the accuracy of feature extraction is insignificant as validated by the results of the ISPRS test on image understanding (Sester et al, 1996). The test showed that there is no significant difference between building extraction results based on normalized DSM and that of analysis of height layers also called bins.

Owing to limitations of the "normalized" DSM approach, this research applies a scheme of raised structure extraction which is based on segmenting the DSM by global height thresholding similar to Paparoditis et al (1999). Other techniques of raised structure extraction from the surface model were tested in this research but proved ineffective.

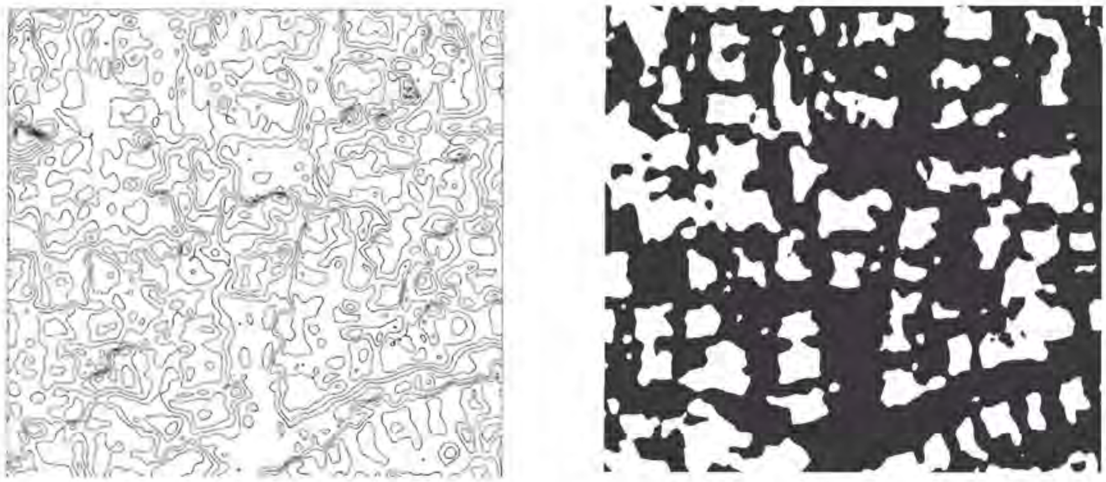
The tested raised structure extraction techniques were:

1. Edge detection in the surface model. This was executed using the Sobel operator and it resulted in a large number of edges from which it was difficult to identify edges which define building edges (see Figure 14a and 15a).
2. Extraction of 3-D edges on the basis of edge magnitude. This did not work well due to small height variations in the surface model i.e. most edges had similar edge magnitude value.

In the DSM global height thresholding approach, height threshold values used are deduced from prior context knowledge of the study areas as well as from buildings to be extracted. For the Manzese Site 1, the generated surface model had heights ranging from 22.8 to 35 m above mean sea level (amsl). It is known that buildings in this area vary in height from 3 to 4 m above the average ground. A global height threshold value of 28 m (amsl) empirically determined, proved effective in detecting most building roofs and was applied. It must be noted that the approach of detecting building roofs by global thresholding of the digital surface model holds only under the assumptions that the terrain is relatively flat and most buildings could detected. In the case of a sloping terrain, a DTM is required for detecting raised structures that are above a global threshold height value. The DTM may be subtracted from the sliced DSM layer thereby distinguishing raised structures from terrain points. Additionally, buildings that are below the global height threshold value can only be detected if the sum of the building's average ground level height and the building's height is above the global height threshold value, i.e.

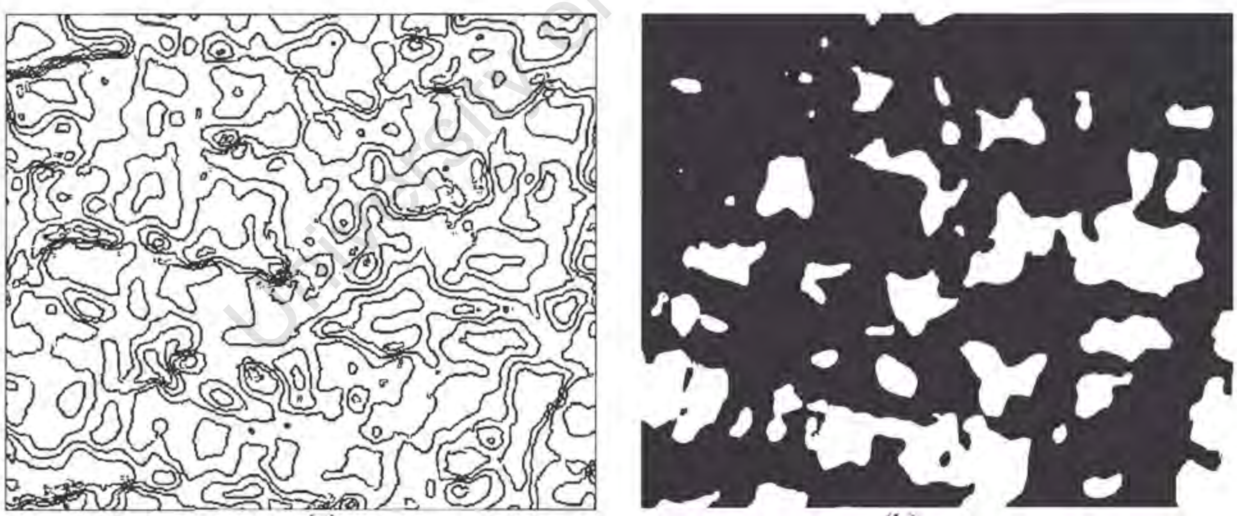
(Average Building Ground Level + Building Height) > (Global Height Threshold Value)

Undetected buildings in the course of DSM global height thresholding technique may be added later either manually or otherwise. For the Manzese Site 2, the generated surface model had heights ranging from 24.3 to 32.4m (amsl), a height threshold value of 29.5m empirically determined proved effective and was applied. The results of global height thresholding of the surface model are binarized raised structure blobs as shown in Figure 14b and 15b.



(a) (b)

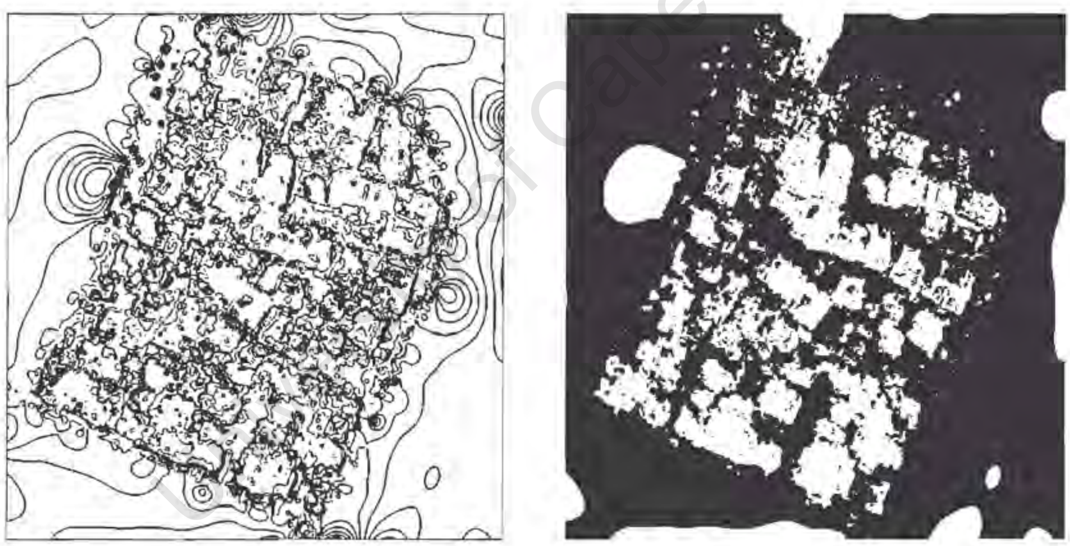
Figure 14. (a) Surface model edge detection (NOTE: the surface model edges were not used in the detection of raised structures) (b) Thresholded and binarized raised structure blobs – Manzese Site 1.



(a) (b)

Figure 15. (a) Surface model edge detection (NOTE: the surface model edges were not used in the detection of raised structures) (b) Thresholded and binarized raised structure blobs Manzese Site 2.

As may be seen from Figure 12c, the Marconi Beam site, is relatively flat though it has heights ranging from 9 to 23.5 m above mean sea level. Buildings are very close to each other and are approximately of the same height (mostly within 2-4 m). Global thresholding of the surface model on the basis of height resulted in highly melted or merged blobs which were difficult to separate for the subsequent individual raised structure extraction process (see Figure 16b for raised structure blobs at a global height threshold value of 17 m). This is because building height differences were relatively small. Such a situation is critical when using surface models in building extraction.



(a) (b)

Figure 16. (a) Surface model edges (Sobel Operator) (b)Thresholded (17 m) and binarized DSM – Marconi Beam site *(white areas are the elevation blobs).*

In the light of the foregoing situation, raised structure blobs in the Marconi Beam area were derived by segmenting the surface model using the Multiple Height Bins (MHB) approach (Baltsavias, Mason and Stallman, 1995). In the MHB method, the surface model heights are grouped into height ranges (called bins). This results in segmenting the DSM in relatively few blobs that are always closed and easy to manipulate. The results are shown in Figure 17. In this approach, only raised structure blobs within selected height ranges can be detected. The approach is usually applied hierarchically where bins of large height ranges are used for coarse detection of raised structure blobs and bins of small height ranges are used to verify and refine the coarse detection. The MHB method can be applied either globally i.e. over the entire surface model or locally i.e. for selected areas. In this research, the MHB technique was applied globally. The maximum and minimum bin ranges were empirically determined. It must be noted that the empirical approach of selecting bin ranges works under the assumption that a considerable amount of buildings are detected by the selected range and that the terrain is relatively flat. Three height bins were used in the Marconi Beam area to detect most building roofs (see Figure 17).

(a) Bin 1- Height >= 18.0

(b) Bin 2 – 17.5<=Height < 18.0

(c) Bin 3 – 17.0<=height <17.5

Figure 17. Binarized multiple height bins – Marconi Beam site.

It is important to emphasize here that the results of individual bins are filtered to remove irrelevant blobs (on the basis of area) before adding them together creating a combined layer (see Figure 17c). Because raised structure blobs are geo-referenced to the same frame of reference as orthoimages, their validation could be assessed by superimposition as shown in Figure 18.

(a) Manzese Site 1 (b) Manzese Site 2

(c) Marconi Beam

Figure 18. Raised structure blobs superimposed on orthoimages.

4:6 SEPARATION OF MERGED BLOBS AND BUILDING HYPOTHESES GENERATION

The purpose of height thresholding or segmenting the DSM is to generate blobs that can potentially represent raised structures in the real world. This process, however, is associated with some problems. For example, it is difficult to distinguish between the raised structure blobs that represent buildings and those which represent other objects such as trees, small mounds and so on. Closely situated raised structures will often be merged or melted together in the resulting blobs. In some cases, the image matching process may fail to model raised structures properly thereby leading to blobs which do not represent raised structures in the real world. Because of these problems, raised structure extraction by DSM segmentation can never be comprehensive in itself. The results of DSM segmentation may thus be used only as a starting point for detailed image processes, for example in feature extraction. Prior to using DSM blobs in feature extraction process, they need to be processed e.g. separating the merged blobs and verifying the ground objects they represent i.e. hypotheses verification. This section deals with the problem of separating merged raised structure blobs.

Merged DSM blobs are the result of the weakness of the image matching process which smoothens out height discontinuities between buildings and the terrain in cases of very close structures. This means that height discontinuities between very close raised structures are poorly modeled. This may be attributed to inadequate DSM grid spacing, occlusions and mismatches. Separation of merged blobs is necessary in order to let subsequent extraction processes focus on the modeling of individual raised structures. The effect of merged blobs is critical to building extraction in informal settlement areas, yet it is uncommon in formal settlement areas (Förtsner and Weidner, 1995). A common solution to the problem of merged blobs is using either binary erosion or shadows. The binary erosion approach is based on the separation of blobs through shape manipulations. The approach may not always be effective because merged blobs are a result of smoothing out of height discontinuities as mentioned earlier, so, it is unsatisfactory to separate blobs on the basis of elevation blob shape which is a derived product. It is desirable to solve the problem right from its cause rather than from blob shapes, which are, derived products. It is thus logical to separate merged blobs on the basis of DSM height manipulations.

The approach of using shadows involves the subtraction of shadows from merged blobs which effectively eats away areas in-between blobs (Mason and Baltsavias, 1997; Mason and Rüther, 1997). This is normally done in the image domain. The approach however, may not work in case of images with poor shadows. In addition to that, even in the presence of shadows which are usually obtained by image thresholding, the approach may only be effective for blobs merged in the direction of shadows (see the position of shadows with respect to blobs in Figure 19c). It can be seen that shadows are aligned in the West-East direction of the page, while merging of most of the blobs is in the North-South direction. The shadows approach has been tested in this research. The results are shown in Figure 19d from which it can be concluded that the approach could not effectively separate the merged blobs and therefore it has not been applied in this research.

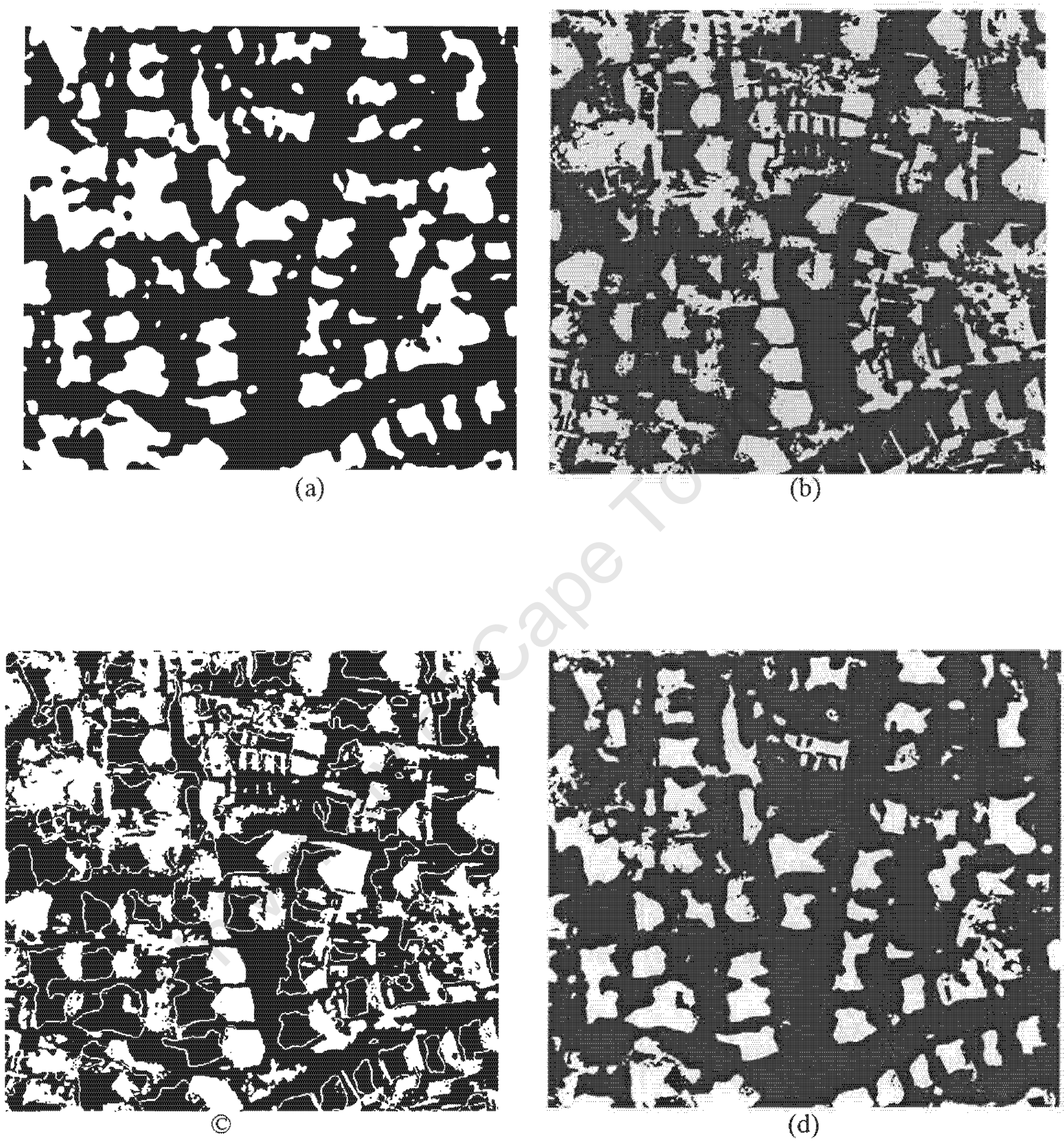
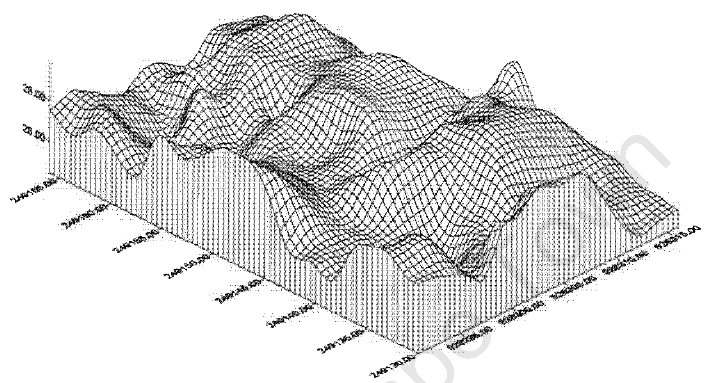
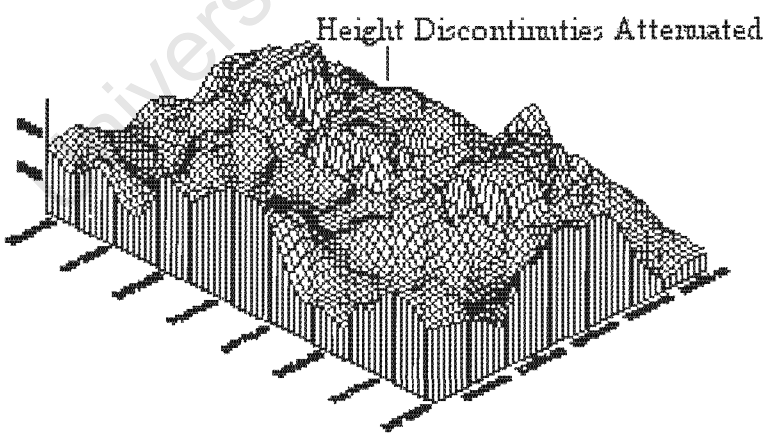


Figure 19. Use of shadows in merged blobs separation (Case Study of Manzese Site 1).
(a) Raised structure blobs (b) Shadows (c) Position of shadows with respect to blobs
(d) Blobs minus shadows.

In this research, merged blobs were separated by using gray scale erosion of the DSM using an appropriate kernel. Gray scale erosion was used to enhance or attenuate height discontinuities as shown in Figure 19 thereby achieving the separation of merged blobs.



(a) 3-D Wire Mesh Surface model of a part of Manzese-Site 1 before gray scale erosion.



(b) 3-D Wire mesh surface Model of a part of Manzese-Site 1 after gray scale erosion.*(note that height differences in between the raised structures have been enhanced)*

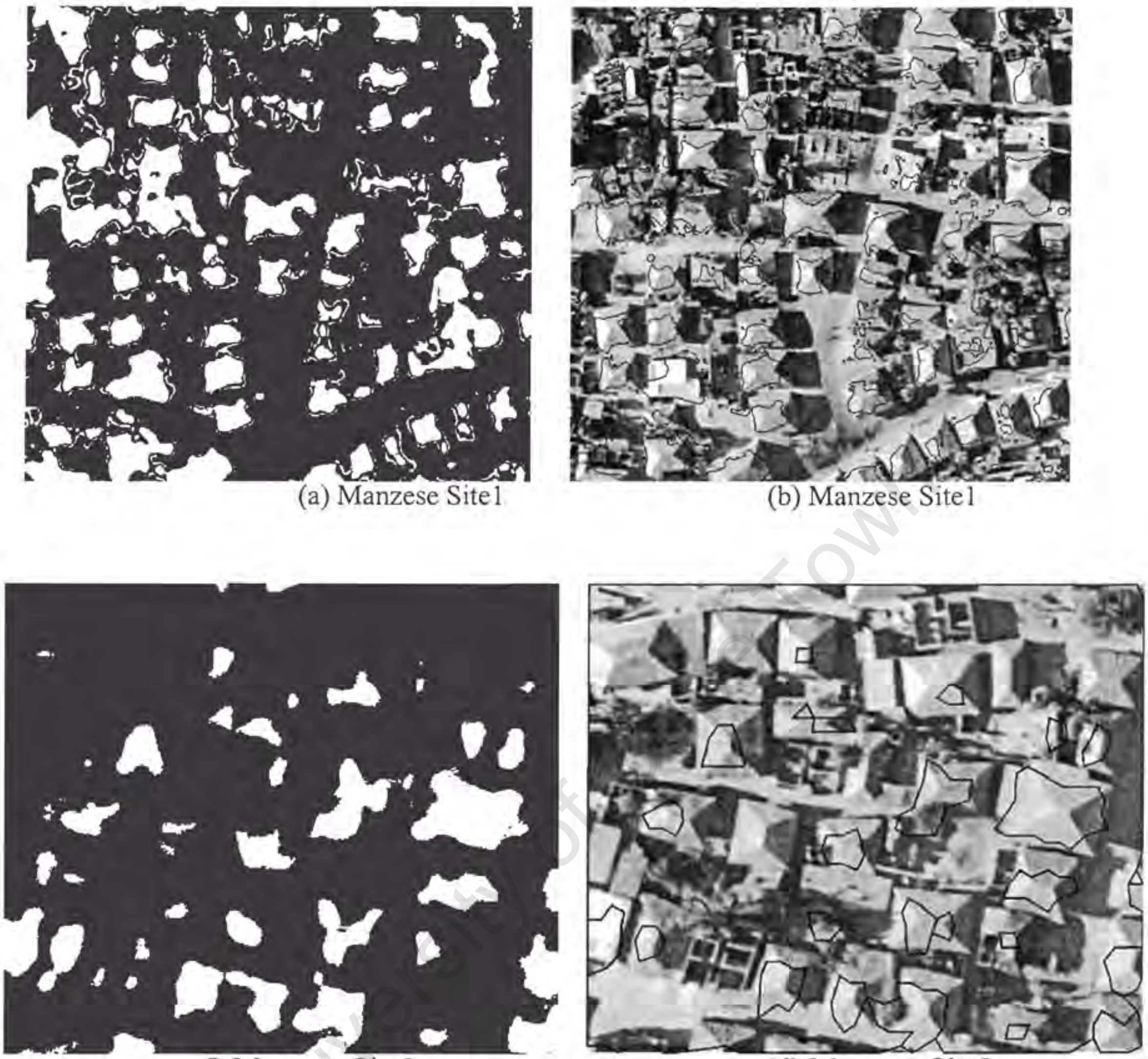
Figure 20. The effect of gray scale erosion of a surface model.

In the gray scale erosion of the surface model, the surface model is convolved with a kernel through which new height values are derived that have height discontinuities attenuated. The design of a kernel is critical in this process, and in most cases kernels are designed after a trial and error process. It should be noted that gray scale morphological filtering kernels operate on the basis of Equations 17a and 17b and not as normal kernels, which are used in routine image processing operations, for example in contrast enhancements, edge detection and so on. In this research, a 3 by 3 kernel with a constant height value of 5 m proved effective and was applied (see Figure 21).

5	5	5
5	5	5
5	5	5

Figure 21. The applied gray scale erosion kernel.*(Note: The origin of the structuring element's frame of reference is at the center pixel)*

Results of gray scale erosion of the surface model and resulting raised structure blobs are as shown in Figure 22.



(a) Manzese Site1 (b) Manzese Site1

© Manzese Site2 (d) Manzese Site2

(a) and (c) Raised structure blobs after gray scale erosion of the DSM (b) and (d) separated blobs superimposed on orthoimages.

Figure 22. Raised structure blobs after gray scale erosion of the surface model. (NOTE: Success rate evaluation is presented in § 5:2:4:3)

It should be noted that DSM gray scale erosion was not applied to the Marconi Beam study area because the raised structure blobs generated from the MHB approach just happened to be not merged (see Figure 18c). Separated blobs in subsequent extraction processing were treated as raised structure hypotheses.

4:7 BUILDING HYPOTHESES VERIFICATION

The purpose of hypothesis verification is to put a label to every raised structure blob that identifies if it represents a building or not. As mentioned in § 1:4, the study areas contain mainly buildings and there is very little vegetation coverage. Because of this, the building hypotheses verification problem in this research was reduced to distinguishing between building and vegetation blobs. Existing solutions to this problem include distinguishing blobs on the basis of shape descriptors e.g. area, perimeter, compactness etc.; use of surface normal variances over the DSM; use of the frequencies of edge directions of orthoimage patches (hereafter called image patches) containing the elevation blob's orthoimage areas and the use of the planar surface fitting to blobs heights.

The approach of applying shape descriptors cannot work well in situations where vegetation areas are almost equal to building areas or when vegetation areas are close to buildings and/or merged together in the surface model or where buildings are of varying shapes. The application of this approach in this work was limited by the fact that buildings in informal settlement areas could not be generalized as having specific ranges of shape descriptors. The approach is ideal for planned areas where buildings are structured and thence have specific ranges of shape descriptors (Weidner, 1997).

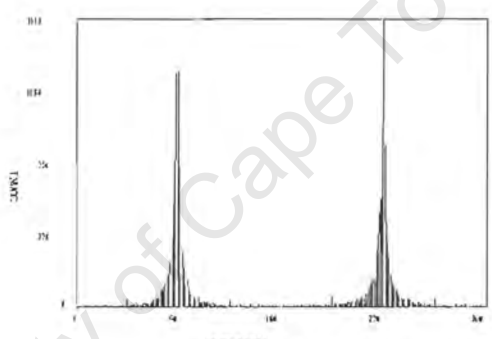
The use of surface normal variances is based on computing surface model roughness by the detection of crease edges (Weidner, 1997). The basic principle behind the approach is that for all pixels of each elevation blob a surface normal is computed followed by a computation of the variance of the surface normal over the blob's DSM area (Hoffman and Jain, 1987). In the case of buildings with nearly flat roofs, surface normal variance is expected to be smaller than that of vegetation areas. The application of this approach in this research was limited by the fact that buildings in the study areas had small roof areas, and are thus covered by limited surface model points, which could lead to inaccurate variance values. In addition to that, the method depends on the quality of the surface model used. The method of using planar surface fitting is most effective in large buildings with flat roofs as pointed out by Paparoditis et al (1998). Buildings in the Manzese study area have ridged roofs with relatively small-areas which could lead to the method being computationally involved. Additionally, the Marconi Beam study area has buildings with flat roofs but with relatively small areas as well. The shape and size of roofs in study areas thus limited the application of the method in this research.

Hypothesis verification in this work has been based on the approach of analyzing frequencies of edge directions in respective orthoimages patches as done by Mason, Baltsavias and Stallman (1997). The principle behind this approach is that for each elevation blob a corresponding orthoimage patch is extracted by fitting a bounding box to the blob. A histogram of edge directions of each orthoimage patch is then computed. It is expected that orthoimage patches that contain buildings will have histograms with high frequencies that are 90° or 270° apart (for regularly shaped buildings) with some additional peaks for more angular buildings. This is because adjacent building edges are generally expected to be at 90° or 270°. On the other hand, histograms of orthoimage patches that contain vegetation areas are expected to be almost flat as shown by Mason, Baltsavias and Stallman (1997). The approach is a simple, fast to implement and accurate method for distinguishing buildings from vegetation and other areas. Examples of hypothesis verification are as shown in Figure 23. In this research, all raised structure blobs proved to be building hypotheses.

(a₁)

(a₂)

(a₃)

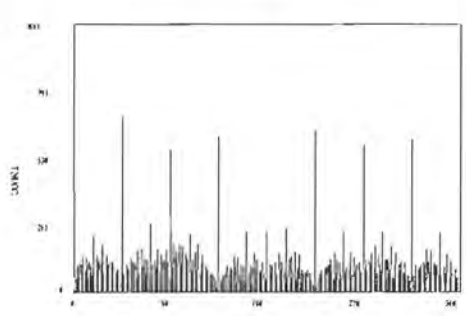


(a₄) Manzese Site 1

(b₁)

(b₂)

(b₃)



(b₄) Manzese Site 2

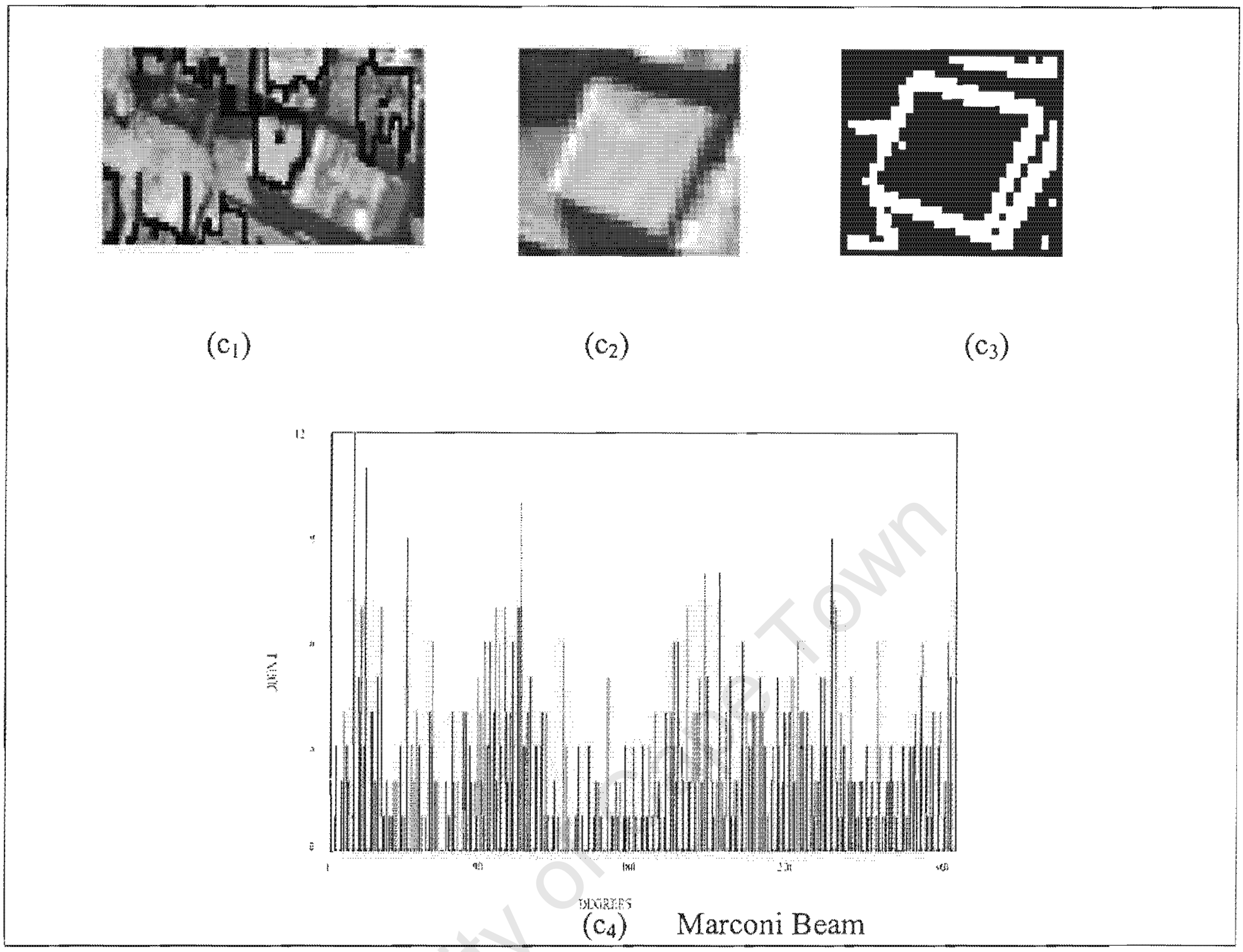


Figure 23a. Examples of hypothesis verification results.

NOTE:

- (a_1,b_1,c_1)-Raised structure blob superimposed on orthoimages.
- (a_2,b_2,c_2)-Extracted part of orthoimage after fitting a bounding box on a raised structure blob.
- (a_3,b_3,c_3)-Edge image of the extracted image patch.
- (a_4,b_4,c_4)-Histogram of edge direction image of the extracted image part. *(Note the edge direction image is not shown in the diagram)*

From Figure 23a it can be seen that the histograms of images containing a building as defined by elevation blobs have high frequencies that are 90° or 270° apart. This implies that the respective elevation blob is contained in a building roof and thereby verifying the hypothesis that the elevation blob represents a building.

4:8 GENERATION OF APPROXIMATE BUILDING CONTOURS

As may be seen from Figure 22 most blobs fall on top of building roofs, but do not precisely delineate roof boundaries. The aim of this section is:

1. To use elevation blobs that are hypothesized as contained in building roofs in defining focus of attention areas in orthoimages. These are areas that are expected to contain or are associated with buildings.
2. From (1) above to initiate a strategy for the extraction of approximate building contours.

The generation of approximate building contours is based on the fact that since blob's centre points typically fall somewhere in the middle of building roofs, it is logical to determine approximate building contours on the basis of the region growing process integrated with edge detection. This is on the assumption that buildings are made up of almost the same materials, as already explained in § 3:5:2.

In segmenting an image into approximate uniform brightness regions, region growing and edge detection processes are normally regarded as being complementary to each other. This means that segments obtained by the region growing process implicitly define edges and vice versa. In view of that, the two processes can be integrated when making a decision whether a point is on an edge or in a homogeneous area (Pavlidis and Liow, 1990). This validates the application of region growing controlled by edges in defining approximate building contours. Approximate building contours are defined using the centroid linkage method of region growing controlled by edges (refer § 3:5:2). This is achieved by scanning from elevation blob centre points in a pre-determined manner such as top-bottom and right-left as depicted in Figure 23b. The effectiveness of region growing depends on homogeneity and contrast in roof areas. Homogeneity criteria parameters could be obtained manually say by the trial and error approach. However, this approach cannot be effective over an entire image because of likely variations of homogeneity in different parts of the image. Aiming at automation, such manual selection of parameters was avoided, instead, homogeneity parameters were computed on a local area basis i.e. for each elevation blob's orthoimage area, prior to the region growing process.

The homogeneity parameters were computed for each elevation blob's orthoimage area through training. Elevation blobs' orthoimage areas (only those hypothesized as contained in building roofs) were used in training to determine parameters through which the region growing process could achieve adequate delineation of complete building roofs (see Figure 23b). Training was done by automatic sampling of pixels in respective elevation blob orthoimage patches from which patch statistics were computed (refer Figure 23b for the sampling format). For each elevation orthoimage patch a total of about 300 pixels was sampled and the following statistics were computed:

- mean pixel value,
- maximum edge magnitude and
- mean pixel value difference from the centre pixel.

These statistics were used in defining homogeneity criterion, by which the rest of the pixels in the proximity of the elevation blobs' orthoimage areas are classified as either belonging to a building roof or not. The training process was followed by actual region growing process to define approximate building contours. This involved the sampling of pixels in each area surrounding the elevation blob's orthoimage area and classifying each as either belonging to the building roof or not. A pixel was declared belonging to a building roof if it satisfied a stipulated homogeneity predicate (as defined from the above statistics). The homogeneity criterion could be on the basis of either edge magnitude, i.e. sampling until a pixel with predetermined edge magnitude is met or pixel value homogeneity i.e. sampling until a pixel with significantly different intensity within user defined limits is met, or on variances from the elevation blob centre pixel. All these criteria were tested and the edge magnitude criterion proved effective and was applied. The technique of region growing on the basis of edge magnitude criterion is referred to in the rest of this thesis as region growing constrained by edges.

It must be reiterated that there were two types of pixel sampling and both were from elevation blob's centre points in the orthoimage. The elevation blob's centre points were determined by averaging the X and Y object space coordinates of blob vertices and projecting them to the orthoimage. The first sampling was for the determination of statistics that best describe each orthoimage patch covered by elevation blobs (those hypothesized as contained in building roofs). This first sampling was for training to establish statistics that are characteristic of an orthoimage roof area covered by each elevation blob. The second sampling was for region growing to determine approximate building regions on the basis of statistics that are obtained from the first sampling process.

The results of region growing constrained by edges were approximate building regions as shown in Figures $24a_1$, $24b_1$, and $24c_1$. The approximate building regions were subsequently vectorized resulting in approximate building contour's coordinates that were inputs into the dynamic programming optimization process.

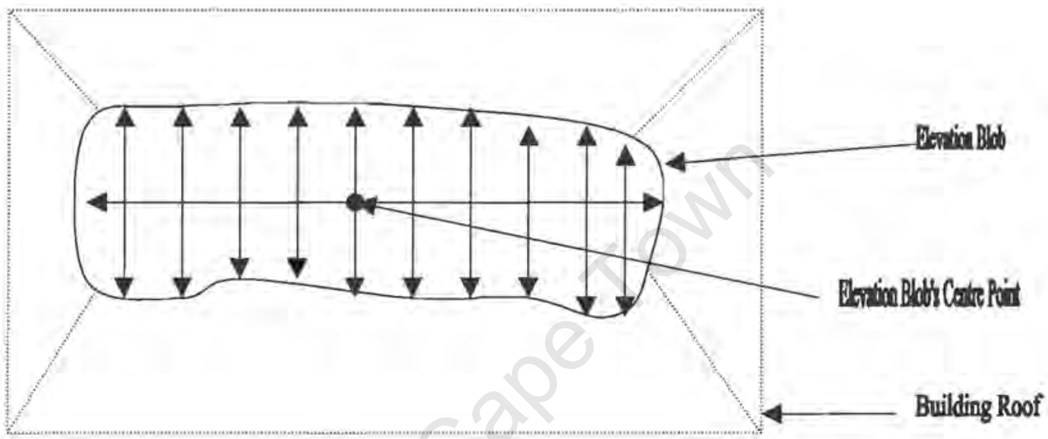


Figure 23b. The scanning format adopted in each elevation blob (hypothesized to be contained in a building roof) to determine roof area image statistics. *(Note: arrows do not indicate scanning termination points, termination could be at a point either beyond or below the arrow)*

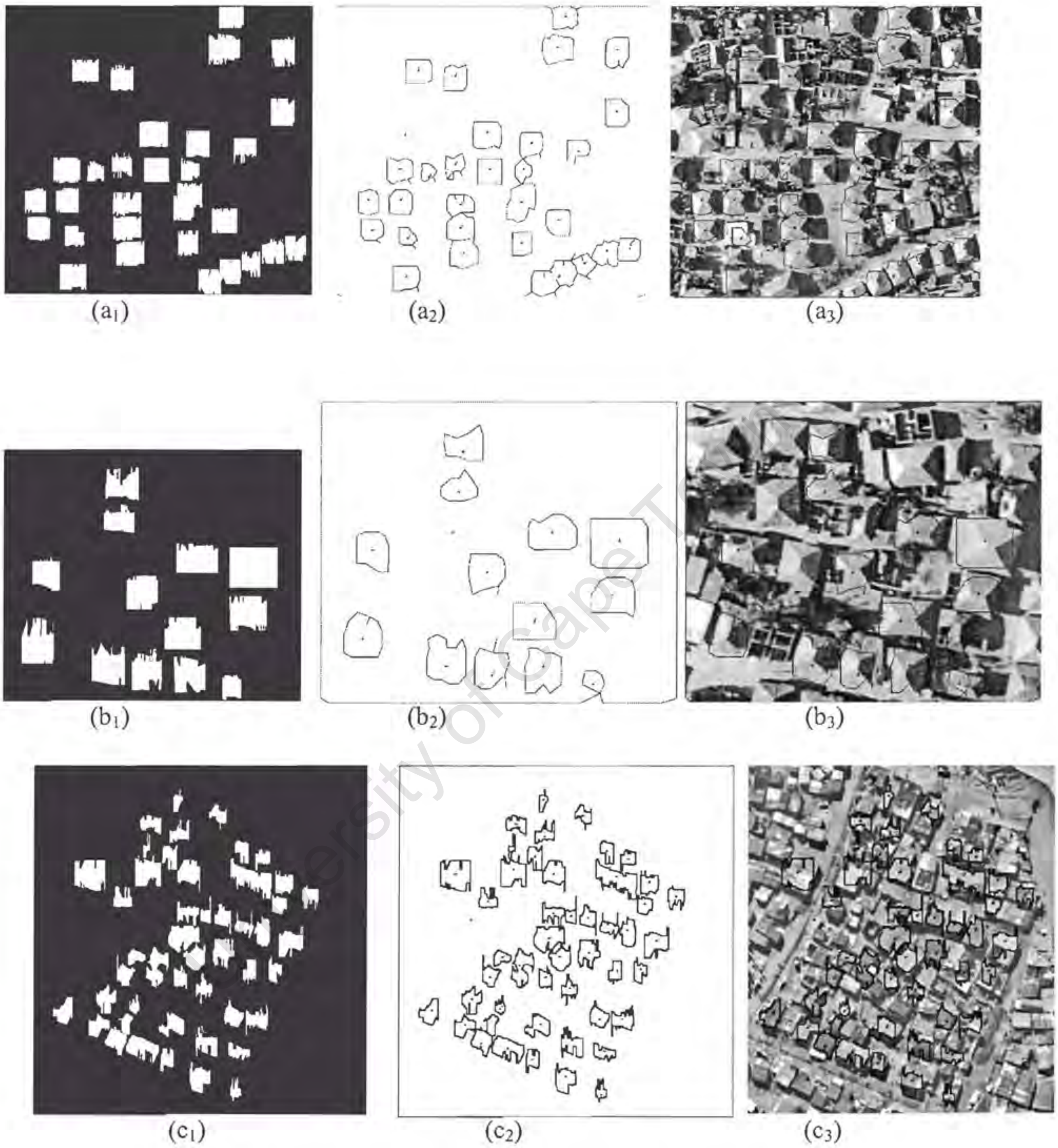


Figure 24. Results of regional growing and approximate building contours.

NOTE:

- (a_1,b_1,c_1)-Region growing and edge detection for Manzese site1, site2 and Marconi respectively.
- (a_2,b_2,c_2)-Approximate building contours Manzese site1, site2 and Marconi respectively.
- (a_3,b_3,c_3)-Building contours superimposed on orthoimages Manzese site1, site2 and Marconi respectively.

4:9 APPROACH TO MODIFICATION OF APPROXIMATE BUILDING CONTOURS

4:9:1 CONTEXT

This section is about designing a strategy for the modification of approximate building contours as obtained from region growing and edge detection processes (refer § 4:8), so that they can optimally delineate corresponding buildings. The modification of approximate building contours is due to the artifacts that results from the application of the region growing constrained by edges process. The approach of using region growing constrained by edges in the derivation of approximate building contours results in edges which are assumed to be building boundaries. The approach may result in either of the following errors as pointed out by Pavdilis and Liow (1990):

1. A boundary may not be identified as an edge i.e. the boundary is incorrectly rejected or missed.
2. An edge may not be a boundary i.e. the boundary is incorrectly accepted.

The occurrence of the first type of error can be minimized by the application of contrast enhancement techniques prior to the region growing process. The probability of the occurrence of the second type of error can be minimized by the use of effective homogeneity criteria, though in some cases this may result in over-segmentation thereby increasing errors of the first type. In general, the region growing process may lead to false boundaries because the homogeneity criterion cannot be fulfilled over all designated pixels in the image. It is thus important to modify the results of region growing and edge detection by applying some form of optimization. For reasons already explained in § 3:2 modification of approximate building contours in this research is carried out using the snakes approach and the dynamic programming optimization technique.

4:9:2 FORMULATION OF APPROXIMATE BUILDING CONTOURS INTO SNAKES

As on explained in § 3:3 snakes are characterized by their ability to slither. In this research, formulation of approximate building contours into snakes is realized by the adoption of the following strategies:

1. Defining building contours by their nodes N, N=(n_1, n_2, n_3, ..., n_n), where n_i=(x_i, y_i) are image space coordinates of node n_i.
2. Allowing building contour nodes to change positions. Contour nodes are free to change positions within a window which is defined at each node. In the coarse of nodes changing positions, building contours slither from one position to the other. By controlling the contour slithering it is possible for the contour to delineate objects of interest (in this research, buildings).

In this research contour nodes are free to change position in a 3 by 3 window, defined at each node. When a window is fitted at a node, it defines the node's window points which the node can move to. For each node, in this implementation there are thus 9 node window points (also referred to as the node's degrees of freedom). Out of the nine node window points, one, which optimizes the building cost function, is to be selected as optimum (refer § 4:9:3:1:1 for definition of a cost function).

Variables (unknowns) in the building contour optimization problem are optimum node window point positions. As the variables of the problem are not inter-related, it is not possible to achieve optimization in only one process. Contour optimization thus has to be carried out sequentially through all the nodes by processing one node at any one stage. In view of that, the problem of building contour optimization is thus susceptible to dynamic programming optimization formalism, through which a criterion for selection of optimum node window points is defined.

In order to apply the dynamic programming optimization technique it is necessary to have a generic building mathematical model and a corresponding objective function to reduce the search space. The generic building mathematical model and objective function formulation are explained in the following section.

4:9:3 GENERIC BUILDING MATHEMATICAL MODEL FORMULATION

A generic building mathematical model is an approximate mathematical formulation, which models the general manifestation of a building in an image (refer § 3:2:1 for the definition). Let ψ be a building region and ξ the corresponding building contour in an image, the following assumptions hold:

- A building is defined by its region ψ and building edges which constitute the building's contour ξ;

- The building contour is a curve or snake ξ in a digital image. Let the digital image be represented by a 2-D function *G(x, y)* which is measurable and has continuos derivatives and
- The building contour ξ has continuous derivatives and a unit vector *n(s)* normal to it.

The building generic mathematical model is formulated using radiometric and geometric building characteristics verbally and mathematically as explained in the following section:

4:9:3:1 BUILDING CHARACTERISTICS

4:9:3:1:1 BUILDING RADIOMETRIC CHARACTERISTICS

1. The building edge is a continuous curve and has a relatively high contrast. This suggests that the squared sum of grey values (or their derivatives in the direction normal to the building edge) along the curve attains the maximum. This may be represented mathematically as:

$$E_1 = \int (G(V(s)))^2 ds \Rightarrow Maximum \tag{4:1}$$

Where G is an image function, $V(s)$ is a vector valued function, which transforms the contour length 's' to point $(x(s), y(s))$ in the image space i.e. $V(s) = (x(s), y(s))$; E_1 is building edges energy value.

2. Building regions are continuous and their pixels as mentioned earlier in § 3:5:2 are almost homogeneous i.e. do not vary very much within smaller areas. Mathematically this is equivalent to:

$$E_2 = \sum_{i=1}^{i=n} \int_{\Delta a_i} [\ G(f(a)) - G_m(\Delta a_i)\ ]^2 da \Rightarrow Minimum$$

(4.2)

Where E_2 is building pixels homogeneity energy value; 'a' is a building area in the object space, $f(a)$ is a vector valued function which transforms building area 'a' to point $(x(a), y(a))$ in the image space; Δa_i is a small building area in the image; $G_m(\Delta a_i)$ is the average value of G over area Δa_i that is:

$$G_m(\Delta a_i) = \int_{\Delta a_i} [\ G(f(a)) / \sum \Delta a_i\ ]$$

[Equation 4:2 means that if regions are identified as building hypotheses in an image, regions containing buildings will bear relatively smaller value of E_2 than the regions without buildings.]

4:9:3:1:2 BUILDING GEOMETRIC CHARACTERITICS

1. In terms of geometry, building edges wiggle in delineating buildings. This implies that building contours bear many sharp turning points i.e. points of high curvature. By the fact that curvature can be approximated by second order derivative along the contour, this property can be presented mathematically as:

$$E_3 = \int (G''(V(s)))^2 ds \Rightarrow Maximum \qquad (4:3)$$

Where E_3 is building edges curvature energy value.

Equation 4:3 means that pixels along building edges have relatively higher second order derivative values than pixels along non-building edges.

2. Contiguous building edges typically are at right angles to each other. A building edge may be defined by its start and end nodes. Mathematically this can be expressed as either :

$(|(\alpha_{bi} - \alpha_{bi+1})| - 90) < T$ or

$(|(\alpha_{bi} - \alpha_{bi+1})| - 270) < T$ (4.4)

Where α_{bi} - is a direction vector of building edge b_i and T-user defined threshold.

The above generic building mathematical model is used in formulation of a building cost function. A cost or objective function (terms used synonymously) is a solvable and implementable transformed form of the generic mathematical model. This transformation needs to preserve all characteristics of the generic model as much as possible. The design of a building cost function is explained in § 4:9:5. Inputs into the cost function are building contour node's energy values computed using the merit function as explained in the following section.

4:9:4 THE MERIT FUNCTION

The position of the snake in image space is defined by energy acting on it. The position of the snake can thus be changed by varying its energy. A merit function is a function through which the snake's position is defined by computing its total energy. The total energy of the snake is a sum of all radiometric and geometric energy acting on it. The merit function as defined by Amini et al (1990) and Gülch (1990) is adopted in this research. Amini et al (1990) and Gülch (1990) defined the total energy of the snake by the following merit function:

$$E_{snake} = \int_0^1 E_{snake}(V(s))ds \qquad (4:5)$$

Expanding equation 4:5 to include internal and external energy forces as explained in § 3:3 the following equation is obtained:

$$E_{snake} = \int_0^1 (E_{\text{int}}(V(s)) + E_{ext}(V(s)))ds \tag{4:6}$$

Further expansion leads to a generalized merit function as:

$$E_{snake} = \int_0^1 (E_{int}(V(s)) + E_{ima}(V(s)) + E_{con}(V(s)))ds \tag{4:7}$$

Where:

- E_{snake} is the total energy of the snake.
- E_{int} are internal forces through which geometric constraints are imposed on the shape of the contour. Gülch (1990b) and Amini et al (1990), expressed E_{int} on the basis of curvature as:

$$E_{\text{int}}(V(s)) = \alpha(s)|V_s(s)|^2 + \beta(s)|V_{ss}(s)|^2 \tag{4:8a}$$

  Where:

  - $V_s(s)$ and $V_{ss}(s)$ are first and second order derivatives respectively at a point at length s along the snake;
  - $\alpha(s)$ and $\beta(s)$ are position dependent parameters or weights that control the shape of the snake. The large value of $\alpha(s)$ relative to $\beta(s)$ forces the snake to be closer to points with small curvature and conversely the large value of $\beta(s)$ will force the snake to higher curvature points.

- E_{ima} - Image radiometric forces that are part of the external forces E_{ext} which attract the snake to features of interest in the image. E_{ima} is a combination of different energy terms which are derived or computed using the image function $G(x,y)$ such as pixel intensities values E_{pix}, edge magnitude E_{edge}, cornerness E_{corner} and so on. By introducing a weight function (W_R) to each source of radiometric energy which signifies its contribution to the object's manifestation in the image, E_{ima} becomes a combination of weighted energy terms which may be expressed as:

$$E_{ima} = W_{pix}E_{pix} + W_{edge}E_{edge} + W_{corner}E_{corner} + \ldots + W_nE_n . \quad (4:8b)$$

Weights can be arbitrarily selected depending on the structural configuration and complexity of the object to be extracted. $E_{pix}, E_{edge}, E_{corner}$ are obtained by computations from image data for example:

E_{pix} - Direct pixel values,

E_{edge} - From application of gradient operators like Sobel or others.

E_{corner} -From image corner point computations Rosenfeld and Kitchen, (1982).

- $E_{con}(V(s))$ -External constraint forces acting on the contour which can either be radiometric or geometric.

4:9:4:1 DISCRETIZATION OF THE MERIT FUNCTION

The merit function (Equation 4:7) which defines the position of a snake in an image is in continuous space, so to use it in contour optimization is ideal, simplifying it by discretization. This is logical because the image data are inherently discrete. Discretization of the merit function (Equation 4:7) is carried out, according to Amini et al (1990), by the evaluation of individual component integrals in the merit function at particular points along the snake i.e. at the nodes. Following Amini et al (1990) approach the discretised form of equation 4:7, becomes:

$$E_{snake} = \sum_{i=1}^{i=n} (E_{int}(V(s_i)) + E_{ima}(V(s_i)) + E_{con}(V(s_i))) \tag{4:9}$$

Where:

-i is node number

-n is the number of nodes in a contour

- s_i is contour length at node i

-$V(s_i) = (x(s_i), y(s_i))$ is position of node i in the image.

-Other terms are as previously defined in Equation 4:7.

E_{ima} and E_{int} in Equation 4:9 are further generalized as follows:

- $$E_{ima}(V(s_i)) = \sum_{j=1}^{j=no} RE_j(G(V(s_i))) \tag{4:10}$$

Where:

-RE- is a radiometric energy term.

-$G(V(s_i)) = G(x(s_i), y(s_i))$ - is the image function being evaluated at node i.

-j-radiometric energy function e.g. edge magnitude or corner function.

-no - total number of radiometric energy functions under consideration.

-n – total number of contour nodes.

- By expressing V_s and V_{ss} in terms of image coordinates, E_{int} in Equation 4:8a can be expanded to:

$$E_{int}(V(s_i)) = E_{int}(x(s_i), y(s_i)) = \alpha(s_i)\{(x(s_i) - x(s_{i-1}))^2 + (y(s_i) - y(s_{i-1}))^2\}$$
$$\beta(s_i)\{(-2x(s_i) + x(s_{i-1}) + x(s_{i+1}))^2 + (-2y(s_i) + y(s_{i-1}) + y(s_{i+1}))^2\} \tag{4:11}$$

-Where: $(x(s_i), y(s_i))$– are node i image coordinates and other terms are as previously defined in Equation 4:9.

By back substitution of Equations 4:10 and 4:11 into Equation 4:9 the discretized merit function is obtained as:

$$E_{snake} = \sum_{i=1}^{i=n} \{\beta(s_i)((-2x(s_i) + x(s_{i-1}) + x(s_{i+1}))^2 + (-2y(s_i) + y(s_{i-1}) + y(s_{i+1}))^2) +$$
$$\alpha(s_i)(x(s_i) - x(s_{i-1}))^2 + (y(s_i) - y(s_{i-1}))^2)\} + \sum_{i=1}^{i=n}\sum_{j=1}^{j=no} RE_j\{(G(V(s_i)))\} + \sum_{i=1}^{i=n} E_{con}(V(s_i))$$

(4:12)

-Where terms are as previously defined in Equations 4:9,4:10 and 4:11.

Equation 4:12 computes the total energy of the entire contour. Energy at individual nodes (e.g. at node s_i) and respective window points can be computed by the following equation which is derived from Equation 4:12 above.

$$E(s_i) = \beta(s_i)\{(-2x(s_i) + x(s_{i-1}) + x(s_{i+1}))^2 + (-2y(s_i) + y(s_{i-1}) + y(s_{i+1}))^2\} +$$
$$\alpha(s_i)\{(x(s_i) - x(s_{i-1}))^2 + (y(s_i) - y(s_{i-1}))^2\} + \sum_{j=1}^{j=no}\{RE_j(G(V(s_i)))\} + E_{con}(V(s_i))$$

(4:13)

By discretization of the merit function, the building contour optimization problem becomes susceptible to discrete multistage optimization processes, so it can be optimized by the dynamic programming technique (Cooper and Steinberg, 1970, Amini, et al, 1990).

4:9:5 DESIGN AND REALIZATION OF THE BUILDING COST FUNCTION

The cost function defines the criteria for the selection of optimal node window points (refer § 5:2:3 for definition of nodes window points). The generic building mathematical model, as described in § 4:9:1, is in a continuous space. To apply it in contour optimization, it is necessary to simplify it by discretization and to formulate a corresponding cost function. In this research, the generic building mathematical model is formulated into a discrete form using building properties as expressed by Equations 4:1 and 4:3 and is expressed into a cost function as follows:

$$TE = \sum_{i=1}^{i=n} E(n_{i-1}, n_i, n_{i+1}) \qquad (4:14)$$

to be implemented under constraint as stated in Equation 4:4.

Where: -TE is the total energy (radiometric and geometric) from nodes n_{i-1}, n_i and n_{i+1}

-(n_{i-1}, n_i, n_{i+1}) are any three consecutive nodes along the contour.

-$E(n_i, n_i, n_{i+1})$ is contour energy computed from the three consecutive nodes.

-i is contour node number.

It must be noted that the building property as expressed by Equation 4:2 is not considered in the formulation of the cost function because it does not contribute in the definition of a building contour and thus it is not applied any further.

The objective function (Equation 4:14) implies that in the implementation of the dynamic programming optimization process, building properties as expressed by

Equation 4:1, 4:3 and 4:4 must be satisfied by any three consecutive nodes along the building contour. This means that optimization needs to be designed in such a way that it is carried out sequentially through all the nodes and that processing at any one particular node (central node) should involve energy contribution from preceeding and succeeding nodes along the contour as indicated by the cost function. The three consecutive nodes approach is adopted because it is easier and more convienient expressing geometry (in terms of curvature) by any three consecutive points along the contour than otherwise.

The merit function (Equation 4:13) and the cost function (Equation 4:14) are used in this research for optimization of the approximate building contours. Energy at contour nodes is computed using the merit function and the cost function is used to select optimum node positions. Implementation starts from the first point of the snake and optimization is carried out through a finite set of stages i.e. ($node_1(s_1)$, $node_i$ (s_i).....$node_n(s_n)$). A decision about optimum positions at each node is made from a finite set of possible alternatives at each node i.e. all possible node window point combinations are considered. From the snake's starting point, internal forces are determined followed by the computation of weighted image forces for the whole image or its processed version and stored in a raster of the same resolution as the image. At the nodes, external forces are computed or interpolated as the need may be.

In implementation, three consecutive nodes are involved at each stage of processing from which window points that optimize the cost function are recorded in the form of a position matrix (refer § 5:2:5 for an example). After every round of computations i.e. through all the nodes, a back tracking searching process is subsequently initiated from the last position matrix, backwards to the first position matrix to determine a window point that globally optimizes the cost function for each node (refer § 5:2 for implementation details). The process is iterative in such a way that after each round of computation, the energy of the resulting building contour is computed. If the resulting energy is less than the previous contour's energy the process is terminated otherwise it continues (see Figure 25 for iteration control flow chart). Before a new iteration starts it may be necessary to densify contour points as the need may arise. An example of the implementation of snakes and the dynamic programming process is as shown in Figure 26. The process results in the 2-D delineation of buildings. The results and further implementation details are covered in Chapter Five.

The limitation of the snake's approach in feature extraction is that the snake's initial position must be as close to the solution as possible, a condition that may often be difficult to achieve in practice. The strength of the approach relies on the possibility of the imposition of geometric constraints at the lowest level to guide the search.

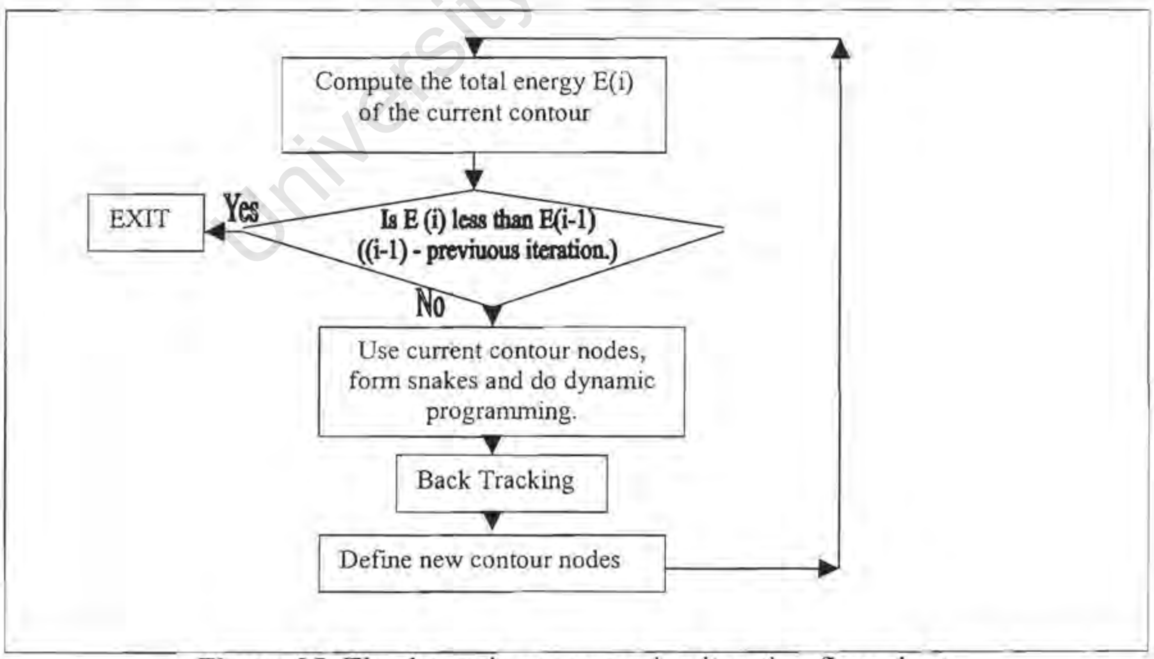


Figure 25. The dynamic programming iteration flow chart.

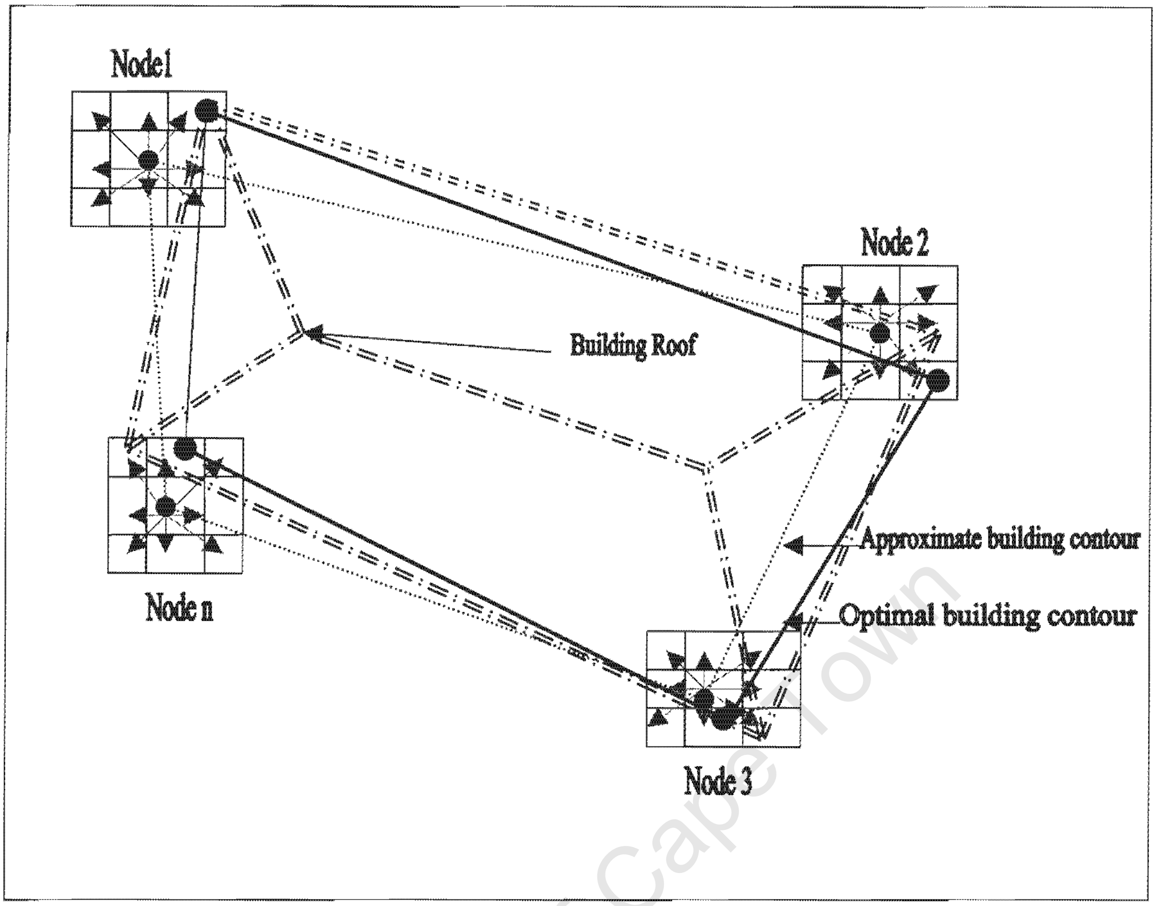


Figure 26. The concept of snakes and dynamic programming optimization technique in 2-D extraction of a building roof. (NOTE: The optimal building contour represents a 2-D delineation of the building)

4:10 APPROACH TO 3-D BUILDING MODELING

Building modeling is the automatic or semi-automatic process of 3-D building reconstruction. Building modeling is useful in the creation of cyber city models and 3-D site visualization. The application of 3-D cyber city models includes telecommunication engineering, flight simulation, urban planning, architecture, environmental engineering, military reconnaissance, rehearsal and maneuvers. Modeling in 3-D is preferable to 2-D modeling mainly because it enables the realistic visualization of features. Visual information is much more easily processed and absorbed by the human brain than numerical or diagrammatic data as in 2-D or 2.5-D representations of our 3-D real world. 3-D modeling creates a great impact on the understanding of information being conveyed (Sheppard, 2000). In informal settlement settings with low literacy rates, visual or graphic communication is a necessary medium in engaging community participation in settlement planning, i.e. communication of ideas and information, thereby allowing residents to discuss scenarios and draw conclusions about developments in their neighborhoods. Currently, there is an increase in the demand for 3-D object modeling to put viewers "into the picture" to help them identify and recognize real world places rather than just viewing abstract maps (Sheppard, 2000).

The current approaches to 3-D modeling of buildings may be categorized into two main schemes. These are:

- Simple visualization schemes – These display the third dimension of the terrain and 3-D building models are draped on a terrain surface using many of the same data structure as conventional 2-D mapping (Maas, 1999; Weidner and Förstner, 1995; Stilla, et al, 2000).
- High-realism (photo-realistic) visualization schemes – These generate a 3-D imagery of the terrain with an almost photographic or ultra realistic nature using techniques such as texture mapping. Building models are usually positioned above the terrain surface.

Simple visualization schemes use library geometric models that are known to best describe buildings in the area of interest. A building model is defined by specific parameters. Building model parameters are usually derived from extracted building contours oftenly using image processing techniques. Model parameters are used in building modeling by using either Constructive Solid Geometry or CAD techniques. Example of such building models are the parametric and prismatic models as described by Weidner and Förstner (1995) and Maas (1999). The parametric building model is known to best describe simple buildings that can be represented by only a few parameters. The parametric building model parameters are described by Maas (1999), as position, length, width, orientation, height, roof orientation and roof steepness. Simple visualization is useful in such applications as telecommunication and so on.

High-realism visualization schemes are preferable to simple visualization schemes as they utilize the benefit of texture and its relationship with geometry in a 3-D graphic solution. High-realism visualization is easy to understand as it supports the user's visual sensitivity.

In this research, 3-D building modeling is a based on high-realism visualization scheme as realized in the ESRI's ArcView 3-D Analyst software package. Results and implementation details are covered in Chapter Five.

4:11 SUMMARY

In this chapter the proposed building extraction method has been explained and partially implemented resulting in initial results. The initial results are the approximate building contour (refer Figure 24a_2, b_2 and c_2). These contours require further processing to generate final 2-D and 3-D building models as described in the following chapter. All processes of the proposed building extraction method can, to a great extent, be implemented in an automated environment. However, processes need to be supported by a human operator who oversees the processes and responds to yes/no type of system queries, sets the required thresholds and assesses the quality of intermediate results for further processing. Detailed implementation issues are covered in the following chapter. It can be noted from Figures 24a_2, b_2 and c_2 that not all buildings are extracted. This is because some buildings were not modeled in the surface model, thence suggesting the need for applying a different extraction strategy. However, more than half of the buildings in each study area have been extracted.

From Figures 24a_3, b_3 and c_3 it can be seen that extracted building contours are close enough to their true building edges. This fulfills one of the snakes technique's pre-conditions which requires that the snake's initial position be as close to its true position as possible. It is thus justifiable to modify the approximate building contours on the basis of the snakes formalism and dynamic programming optimization.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5:1 CONTEXT

This chapter discusses critical implementation issues, reports on 2-D building extraction results, 3-D building reconstruction and evaluates the proposed building extraction system.

The implementation of the proposed system was effected by using both standard and non-standard image processing techniques. Non-standard image processing tasks were implemented by using self-developed computer programs. Computer programs developed are in standard C language. The main program was for the formulation of approximate building contours into snakes and optimization using the dynamic programming technique **(snake.c)**. Other programs developed were supporting programs for the following processes:

- Global altimetric thresholding of the digital surface model **(dtmthresh.c),**
- Gray scale erosion of the digital surface model to separate merged elevation blobs and generation of approximate DTM **(dtmerode.c),**
- Extraction of region growing parameters for elevation blobs that were hypothesized as contained building roofs **(authresh.c),**
- Region growing from centre point of each elevation blob (hypothesized as contained in building roofs) to determine approximate building contours **(grow.c),**

- Approximate building contour simplification **(polymerge.c)**,

The rest of the implementation processes were standard image processing procedures, which were realized using the following application software packages:

- ARC-INFO 7:1:1,
- ILWIS 2.2,
- ERDAS IMAGINE 8:3:1,
- ARC VIEW and
- ARCVIEW 3D ANALYST.

A systematic sequence of the entire processing steps with respective computer programs and application software packages used is shown in Figure 27.

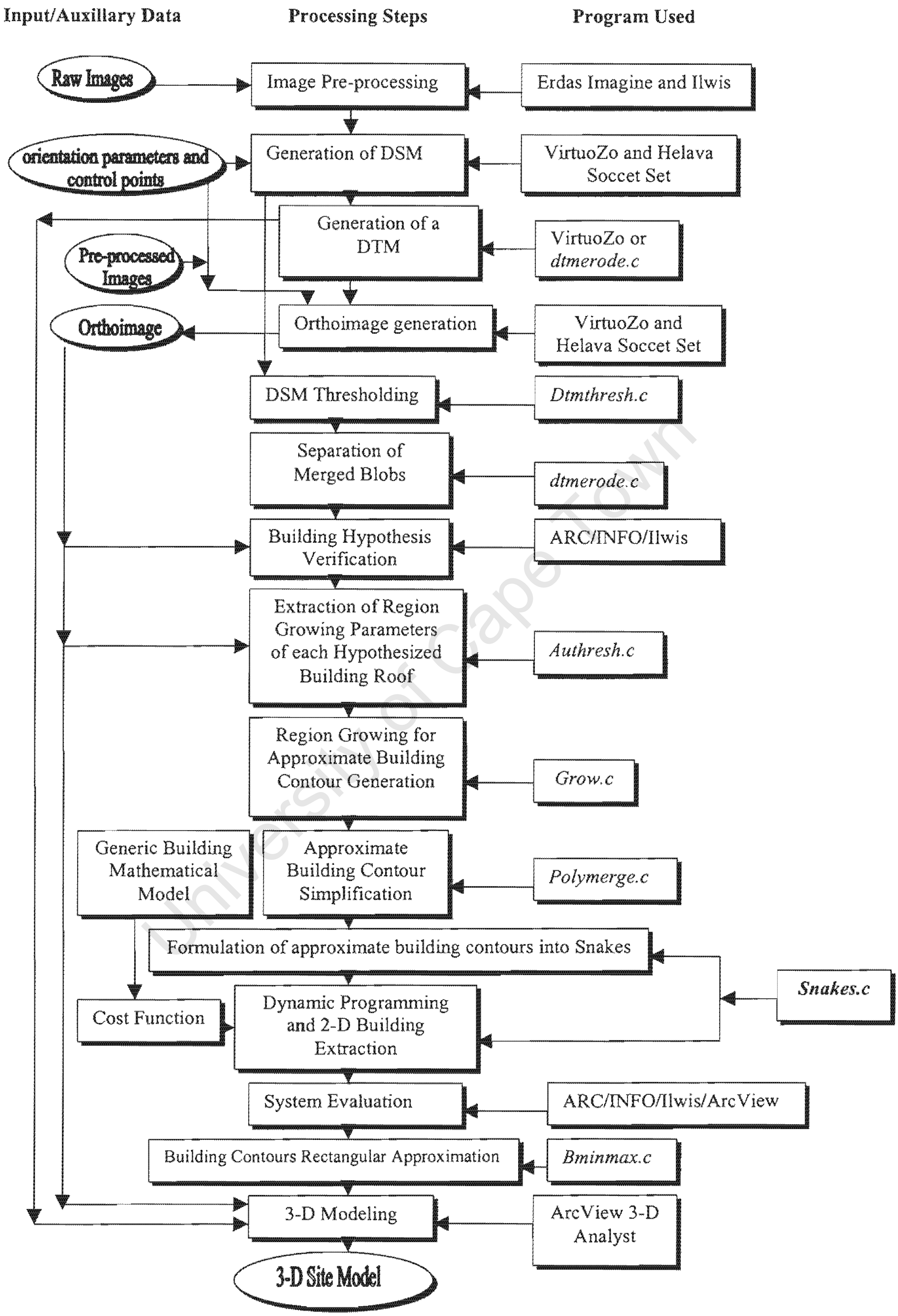


Figure 27 Schematic diagram showing the flow of entire the processing steps and programs used.

5:2 2-D BUILDING EXTRACTION IMPLEMENTATION ISSUES

Final 2-D building extraction, as already mentioned, was carried out by the optimization of approximate building contours as obtained by using the region growing process constrained by edges (refer § 4:8). Modification of building contours was necessary so as to generate new contours that optimally delineate corresponding buildings. Each building contour was processed independently. Critical issues prior to the optimization process were:

1. Simplification of approximate building contours
2. Snakes formulation particularly, the choice of an effective window size for definition of contour node's degrees of freedom,
3. Choice of window point spacing,
4. Realization of the dynamic programming process,
5. Definition of energy terms i.e. functions RE_j as stated in the merit function (Equations 4:12), as well as any other constraints which were deemed necessary to be imposed on building contours in the course of optimization
6. Control of iterations and the back tracking process to determine optimal building contour positions and
7. Densification of contour nodes after each iteration.

Details of all the above issues are explained in the remainder of this chapter.

5:2:1 APPROXIMATE BUILDING CONTOURS SIMPLIFICATION

Approximate building contour coordinates (already explained in § 4:8) were obtained by vectorization. The vectorization process is usually based on connected component analysis which results in the generation of large a number of contour points that could cause computational explosion in terms of memory space, and time during the dynamic programming optimization process. It was therefore necessary to carry out approximate building contour simplification or data thinning prior to the optimization process.

To reduce the number of approximate building contour points, points that seemed to fall along straight lines were eliminated first. This was done on the basis of changes in bearing between consecutive contour line segments. That is, if the bearings between two successive lines segments are equal within some limits, then the three nodes defining the line segments are regarded as being collinear and the central node is thus eliminated. This was followed by further contour simplification in order to remove unnecessary turning points. Existing methods of polygon or contour simplification include the S-Θ plot, chain codes and Fourier descriptors (Ballard and Brown, 1992; Gortschalk and Mudge, 1988).

Out of the contour simplification methods mentioned above, the S-Θ plot was tested. This approach involves the analysis of a graphical representation of contour line's directional vectors versus contour-length. In this representation, the ordinate is the angle of the tangent to the contour at any point of interest measured relative to a fixed direction (the vertical direction was used in this research). The abscissa is the contour-length measured from some arbitrary point of reference on the contour to the point of interest. The contour is represented at uniform contour-length intervals. What is required is the extraction of the contour's critical points from the S-Θ plot. Critical points are points of maximum absolute curvature on the contour. Automatic identification of critical points is extremely difficult. One approach of critical point identification is differentiating the S-Θ plot with respect to the contour length. However, the approach is not always effective because the derivative operator tends to amplify the high frequency noise. Instead, other techniques like the use of multi-scale critical point detector are used (Gortschalk and Mudge, 1988). The method is computationally involved particularly when considering the number and complexity of building contours to be processed in the study areas. Thus, the method was not applied in this research.

Polygon simplification in this research was done by a simple method of successive elimination of nodes (Weidner and Förstner, 1995). The method is applied sequentially through all the contour nodes. Three successive contour nodes i.e. n_{i-1}, n_i, and n_{i+1} are processed at any one stage. The principle behind the method is that any three successive contour nodes will form a triangle; if the formed triangle's base height (h_i) is less than a user defined minimum height value, then, the central node is deleted (refer Figure 28). For example in Figure 28, if h_i is less than a minimum height value, then node i is deleted. For the method to work, the minimum height of a triangle must have a threshold value. A minimum triangle height of 0.5m proved effective in this research and was applied.

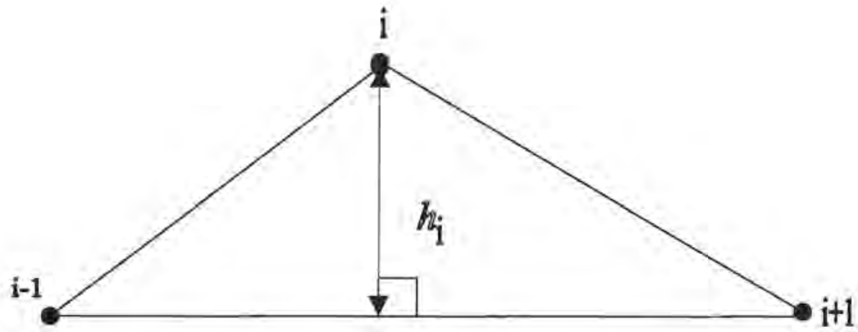


Figure 28. The building contours simplification strategy.

5:2:2 SNAKES FORMULATION ISSUES

As mentioned earlier on in § 4:9:2, snakes were realized by letting the contour nodes change positions within a window in the course of the optimization process. Important issues to consider in the formulation of snakes are the window size and window point spacing as explained in the following section.

5:2:2:1 WINDOW SIZE AND WINDOW POINT SPACING

The change in the position of a node was limited to a window defined at each node. When a window is defined at a node, node's window points are generated at a stipulated interval called window point spacing (refer Figure 29). The window size and window point spacing needs to be determined prior to the optimization process. This is because they guide the process of contour slithering. The number of window points at a node depends on the window size and window point spacing.

Applying a large window with small window points spacing will end up with a large number of window points per node, which results in increased computational efforts.

It must be noted that node movements need to be at sufficiently small distance intervals so that pixels closer to the node take part in the processing because they are more probable than far away pixels. In addition to this, the application of a large window increases the node's movement space, thereby enabling nodes to move freely to far away pixels which may not be part of a feature being delineated. This may lead to erroneous results as shown in Figure 32c. In Figure 32, the window point spacing is fixed at 0.5m and the window size is varied. By comparing Figures 32a, 32b and 32c, it may be seen that optimal delineation of the building (i.e. Figure 32a) is achieved when a 3 by 3 window is adopted.

At a fixed window size, large window point spacing will lead to less computational efforts but increased node's movement space, which once again may lead to erroneous results as shown in Figure 33d. Contrary to this small window point spacing will lead to an increase in computational efforts. In Figure 33, the window size is fixed and window point spacing is varied. By comparing Figure 33a, 33b and 33c, it can be seen that optimal delineation of the building (i.e. Figure33a) is achieved at a window point spacing of 0.5 m.

From the above discussion it can be deduced that the node's movements in a window need to be controlled in some way, so that the points most likely to be feature points take part in the optimization process. In selecting the window size and window point spacing it is necessary to take into account the following factors:

- image spatial resolution,
- spacing in-between the buildings and
- computational efforts resources available.

Window size and window point spacing values are critical in the formation of snakes and in the dynamic programming optimization process. No methods of direct determination of these values exist however, they can be determined empirically prior to the contour optimization process.

In this research, buildings in all the study areas had an average spacing of 1-2 m and at orthoimage's spatial resolution of 0.06 m for the Manzese sites and 0.20 m for the Marconi Beam area, a 3 by 3 window at spacing of 0.5 m empirically determined proved effective, and was applied.

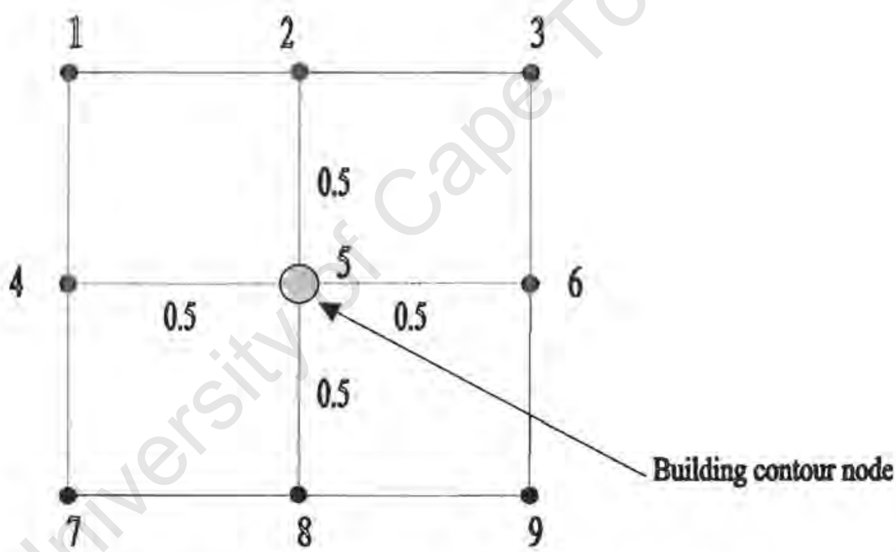


Figure 29. The diagram showing a building node and respective window points as defined in a 3 by 3 window at 0.5m spacing.

5:2:3 DYNAMIC PROGRAMMING IMPLEMENTATION ISSUES

The optimization of building contours, as already mentioned, was effected by the dynamic programming technique. For reasons stated by Amini, et al, (1990) and reiterated in § 3:2:1, variational calculus methods were not applied. Additionally, it was difficult to formulate a functional model that could express the relationships between all variables of the building extraction problem (i.e. the optimum node points). Indeed the number of observations were limited i.e. one approximate contour per building. In view of that, either the exhaustive search or dynamic programming methods were the only feasible solutions to this problem. This is because the problem of optimization of building contours by the snakes approach assumes a finite number of solutions (Cooper and Steinberg, 1970). Additionally, the problem of building contour optimization cannot be solved in one step, thus a multistage approach was the only feasible alternative. Dynamic programming is preferable to the exhaustive search method because it is computationally relatively less involved.

To justify the application of dynamic programming instead of the exhaustive search method in the optimization of approximate building contours, the following analysis will be used. In the course of the snakes formulation let a square window of size 'm', be defined at each building contour node. There will be m^2 degrees of freedom defined at each of the three consecutive contour nodes being processed at one time. There will then be $(m^2)^2 = m^4$ number of possible window point combinations at each central node's window point. This is equal to the number of objective function computations that are to be executed per the central node's window point. The total number of computations for all window points at each central node will then be $(m^4)(m^2) = m^6$. The number of computations for all nodes in a contour will thus be $m^6.n$ (where n- number of nodes in a contour). If m=3 (i.e. 9 degrees of freedom as defined in a 3 by 3 window as applied in this research) there are will be $(3^2)^2 = 81$, number of computations per central node window point (refer Figure 30). Out of the 81 possible window point combinations, combinations which maximize the objective function at the current central node's window point, are recorded in a position matrix (refer § 5:2:5). Thus, for all window points defined at each central node there will be $(3^4)9 = 729$ total number of objective function computations that are to be executed at each central node. If 'n' is the number of nodes in a contour, then there will be a total of $m^6n = 729.n$ number of objective function computations for the entire contour, of which 9.n take part in the analysis for selection of an optimum contour.

There would equivalently be $(m^2)^n$ number of computations by the exhaustive enumeration optimization approach. For example, if m=3 and n=8, by the dynamic programming approach, $8(3^6) = 5832$ computations will result, compared to $(3^2)^8 = 43046721$ by the exhaustive method. In the exhaustive enumeration method, computations increase exponentially and, this favours the dynamic programming approach. However, the higher the number of nodes in a contour the more computationally complex the dynamic programming optimization becomes.

Dynamic programming optimization is a cumulative process (refer Equations 3:3 to 3:7). This arises from the principle that an optimal solution needs to satisfy the fact that whatever the initial decisions are, the remaining decisions in a multistage optimization process must be optimal with respect to the outcome, which results from the initial decisions (Nemhauser, 1996). This ensures a global optimal solution. In view of this, energy accrued from the initial optimization stages is carried forward influencing further optimization stages. For example, processing at one stage i.e. at a particular central node uses energy values computed from respective preceding and succeeding nodes. Then, in the subsequent processing step, a node that was succeeding in the previous step becomes the central node of the current processing step and its energy is taken as accumulated from the previous computational step and carried forward. For the first node of the contour the central node's window points were assumed to have a cumulative energy of zero. For the rest of the central node's window points energy was accumulated and carried forward in the succeeding optimization processes.

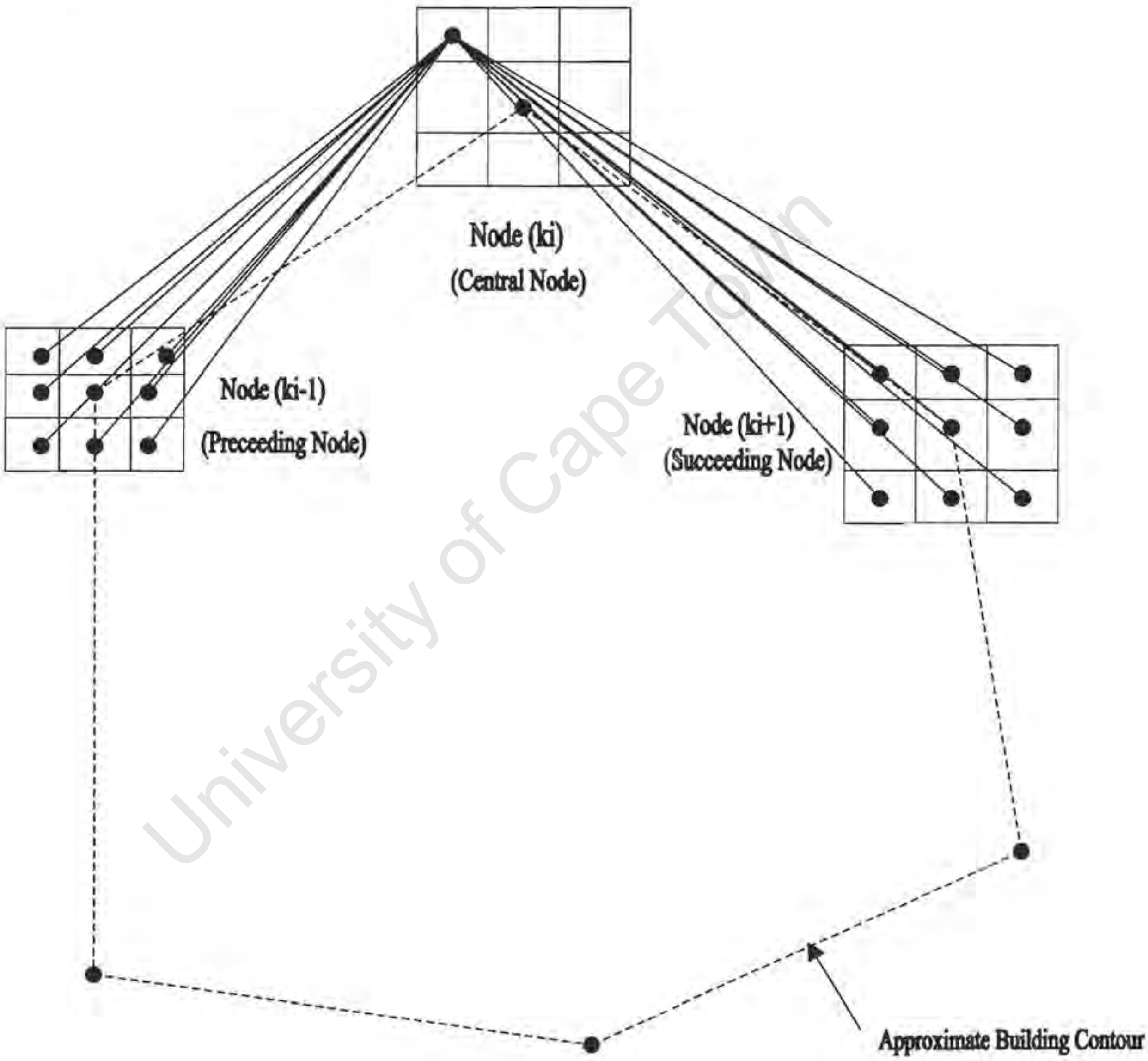


Figure 30. The number of possible window point combinations for one window point of a central node *(dotted lines represent approximate building contour lines).*

For each window point at a node, radiometric energy value i.e. E_{ima} is computed. For the central node's window points, in addition to radiometric energy value computations, geometric energy values i.e. E_{int} is computed for all possible node's window point combinations using Equation 4:11. Node's window point combinations that optimize the objective function at each stage of optimization were recorded in a position matrix (see Figure 34b). A global optimal position of the contour is obtained by searching through all the position matrices. This process is called back tracking and is explained in § 5:2:3:1:4. Applied energy values are explained in the following section.

5:2:3:1 APPLIED ENERGY TERMS

5:2:3:1:1 RADIOMETRIC ENERGY TERMS

The basis for the choice of radiometric energy terms are the building radiometric characteristics (refer § 4:9:1:1:1) which generally presume that a good building edge should have a high cumulative edge magnitude and high contrast. Radiometric or photometric energy terms which have been used to constitute term RE_j in the merit function (Equation 4:13) are pixel edge magnitude and intensity values. That means j=2 in Equations 4:13 which, leads to RE_j being expressed as:

$RE_1 = W_1 E_{magnitude}$; $RE_2 = W_2 E_{pixval}$

$$RE = W_1 E_{magnitude} + W_2 E_{pixval} \qquad (5:1)$$

Where W_1 and W_2 are weights.

The application of Equation 5:1 means that for each contour node and for all points defined in a window around it, edge magnitudes and intensity values have to be computed so that they can be used in the determination of RE_1 and RE_2 values. It should be noted that other radiometric energy terms that could contribute to the value of RE_j e.g. cornerness and second order derivative edge magnitudes were tested and proved ineffective. That is, their contribution to term RE_j proved insignificant and thus were not applied in this research.

5:2:3:1:2 GEOMETRIC ENERGY TERMS

Geometric energy forces are meant to control the shape of a building contour. Curvature is the only geometric energy force that has been applied in this implementation. Curvature was computed from first and second order derivative values at nodes along the contour as already indicated in Equations 4:8a and 4:11. By the back substitution of term RE (Equation 5:1) into the merit function (Equation 4:13), expanding the function and ignoring the last term because it is not explicitly used in this implementation, the following expression is obtained:

$$E(s_i) = \beta(s_i)[(x(s_{i-1}) - 2x(s_i) + x(s_{i+1}))^2 + (y(s_{i-1}) - 2y(s_i) + y(s_{i+1}))^2] +$$
$$\alpha(s_i)[(x(s_i) - x(s_{i-1}))^2 + (y(s_i) - y(s_{i-1}))^2] + \lambda((W_1 E_{magnitude}(s_i) + W_2 E_{pixval}(s_i)) \tag{5:2}$$

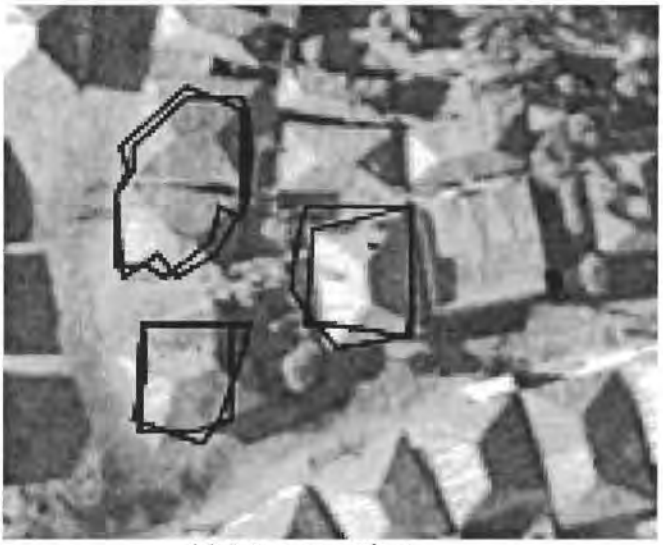
Where: λ is the normalization factor to radiometric energy terms which facilitates the integration of radiometric and geometric energy terms in the merit function. The remaining terms are as defined previously in Equations 4:11 and 5:1.

Equation 5:2 is the merit function, which together with the cost function, (Equation 4:14) has been adopted in this research for the optimization of approximate building contours. Equation 5:2 implies that the energy of a building contour node is the sum of radiometric energy (in this implementation it means edge magnitude and pixel values) and geometric energy which is expressed as a curvature at the node. First and second order derivative values for computation of curvature were obtained by using node image coordinates while edge magnitude and pixel values were extracted and/or deduced from the image data. It must be noted that in the optimization process unit weight was assigned to radiometric energy terms i.e. W_1 and W_2 in Equation 5:2. This means edge magnitude and pixel intensity values were assumed to be of equal weights. This was after tests had proved that there was no significant improvement in building delineation by the introduction of different weight values to radiometric terms. The normalization factor λ was arbitrarily selected and the value of λ=0.5 proved effective and was applied. To impose geometric energy, weights were assigned to first and second order derivative magnitude values. Second order derivative values were assigned relatively high and constant weight value i.e. $\beta(s_i)$ large relative to $\alpha(s_i)$. This was necessary to give priority to high curvature points along the contour which are critical in defining the geometry of a contour. Constant values of 1.0 and 0.5 were respectively used for all $\beta(s_i)$ and $\alpha(s_i)$. The imposition of rectangularity constraint as stated in Equation 4:4 ended with many erroneous delineation of buildings. The constraint was therefore not applied.

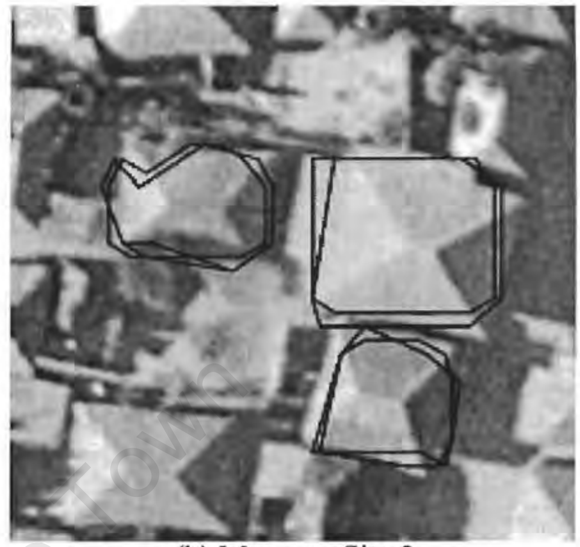
5:2:3:2 CONTROL OF ITERATIONS AND DENSIFICATION OF CONTOUR NODES

Iterations were controlled by the comparison of the contour's energy after every iteration. The end of an iteration is followed by a back-tracking process, through which the energy of the current optimal contour position is computed (refer § 5:2:3:1:4), and compared with the energy of the previous optimal contour. If the current optimal contour's energy is less than the previous optimal contour's energy, the iterations are stopped, otherwise the iterations continue. This is due to the fact that since optimization is based on maximizing the objective function, the global optimal contour position must bear the maximum energy. The global optimal contour needs to be the best out of all possible optimal contours in optimizing the objective function. The geometrical configuration of a resulting contour is evaluated on the basis of its total energy. In this study, an average of five iterations was executed in each building contour (see Figure 31). Figure 31 shows the results of optimization from two iterations. It can be noted that the positions of the contour vary in-between the iterations.

Before a new iteration starts it was necessary to decide whether or not to densify the contour nodes as done by Gruen and Li (1997) in road extraction. The option of contour node densification was tested and proved not to be useful because adding a node along a building contour line effectively does not change the general shape of the contour, and the added nodes only increase the computational efforts. Densification of contour nodes can be effective if the nodes being added are not along the contour edges but are away from them so that the resulting shape of the building contour changes. This alternative was not explored and therefore densification of contour nodes was not applied.



(a) Manzese Site1



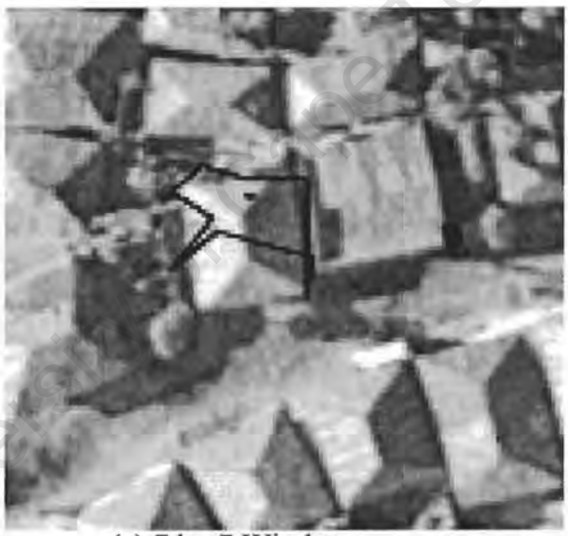
(b) Manzese Site 2

(c) Marconi Beam

Figure 31. The effect of iterations in dynamic programming optimization (Note: Only iterations one and two are shown in the diagram)

(a) 3 by 3 window

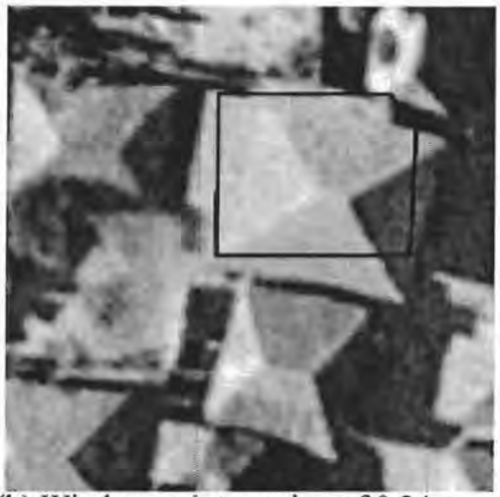
(b) 5 by 5 window



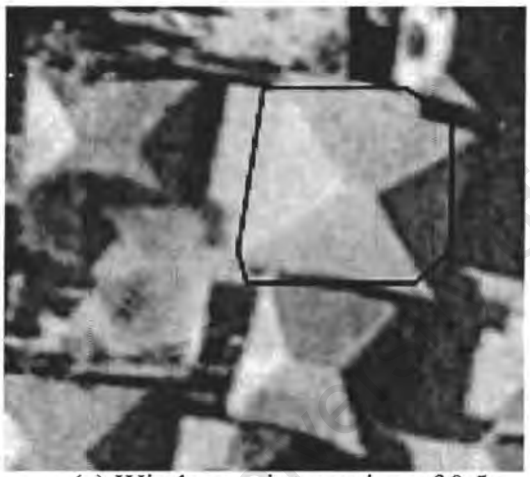
(c) 7 by 7 Window

Figure 32 The effect of window size in the delineation of a building. (Note: By comparing (a), (b) and (c), it can be seen that by fixing the window point spacing (at 0.5) and varying the window size, optimal delineation of the building was achieved when a 3 by 3 window is applied i.e. (a)).

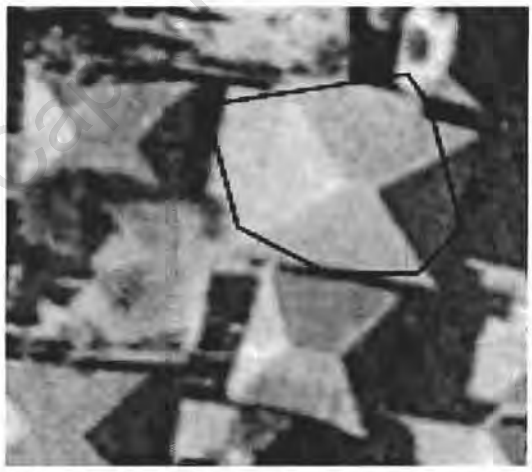
(a) Window point spacing of 0.06m



(b) Window point spacing of 0.24m



(c) Window point spacing of 0.5m



(b) Window point spacing of 1.0m

Figure 33. The effect of window point spacing in a 3 by 3 window in the extraction of a building. (NOTE: By comparing (a), (b) and (c) it can be seen that by fixing the window size and varying the window point spacing, optimal delineation of a building was achieved in (c) i.e. at a window spacing of 0.5 m)

5:2:3:3 THE BACK-TRACKING PROCESS

This is a process of finding a global solution from several local optimum solutions. It is also called a back-ward method of solution. The following example shows how back-tracking is done.

- Let Figure 34 be an example of a building contour with four nodes, i.e. N1, N2, N3 and N4. Let us assume that one iteration of the dynamic programming optimization process has already been carried out to the building contour by fitting a 3 by 3 window to each of the nodes.

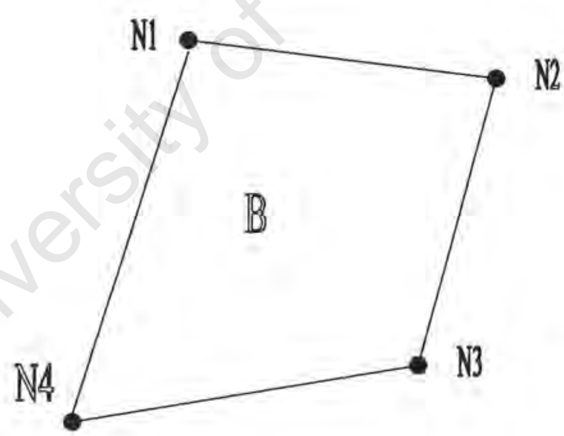


Figure 34. An example of a building contour.

Let the Tables in Figure 34b be position matrices for the iteration. Back-tracking is carried out from the last position matrix by following the arrows as indicated in Figure 34b. In position matrices (Figure 34b), the first column represents the central node's window points that best satisfy the objective function at the current stage of processing; the second column is the corresponding maximum contour energy value and the third column is the corresponding succeeding node's window point.

N1	Energy	N2
2	150	1
2	140	2
4	145	3
5	125	4
6	120	5
7	140	6
9	146	7
2	155	8
3	130	9

Position matrix at node N1

N2	Energy	N3
4	50	1
5	55	2
4	60	3
8	70	4
2	60	5
1	55	6
6	90	7
7	80	8
9	85	9

Position matrix at node N2

N3	Energy	N4
2	200	1
3	150	2
4	170	3
2	205	4
3	175	5
3	185	6
4	179	7
5	200	8
1	203	9

Position matrix at node N3

N4	Energy	N1
3	120	1
2	110	2
6	100	3
7	100	4
8	105	5
2	125	6
3	130	7
4	95	8
6	115	9

Position matrix at node N4

Figure 34b Examples of position matrices at the end of an iteration.

The global optimal position for the building contour above (Figure 34) is established by back-tracking (following the arrows in Figure 34b) i.e. from the last position matrix to the first. For the above example, the global optimal position of the contour is established as being through the following node's window points:

- At node 4 – window point 3,
- At node 3 – window point 4,
- At node 2 – window point 8 and
- At node 1 – window point 2.

In the above example, the total energy of the global optimal position of the building contour is computed as being the sum of energy from all optimal window points i.e. 130+170+70+155 = 525. The global optimal contour is as shown in Figure 35 below.

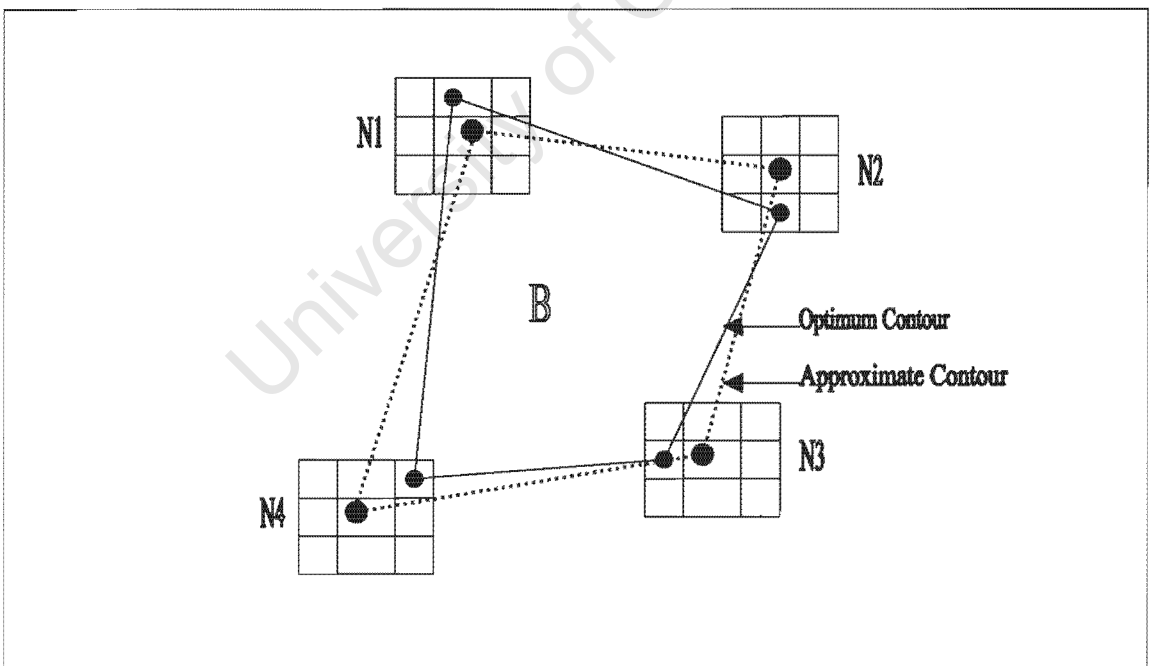
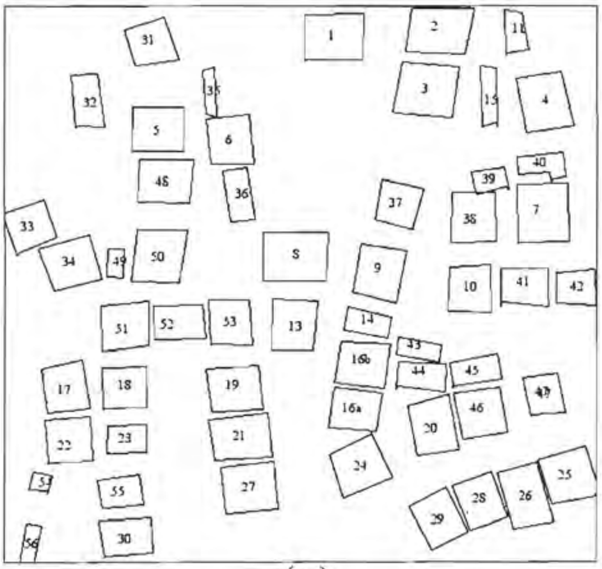


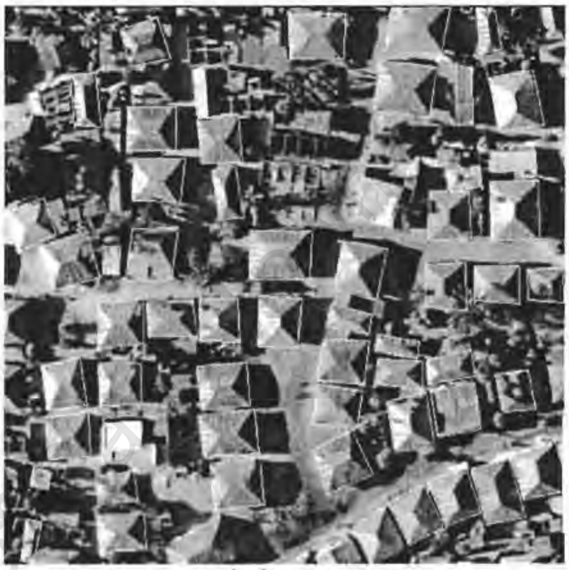
Figure 35. The position of the optimum contour in relation to the approximate contour.

5:2:3:4 2-D EXTRACTION RESULTS

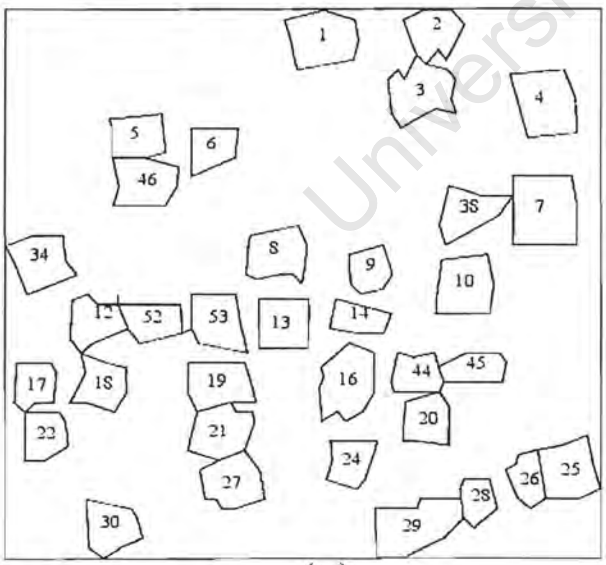
The results of 2-D building extraction are building polygons (in vector form) as shown in Figures 36a$_3$, b$_3$ and c$_3$. Resulting building polygons are validated by superimposing them on respective orthoimages (see Figures 36 a$_4$, b$_4$ and c$_4$).



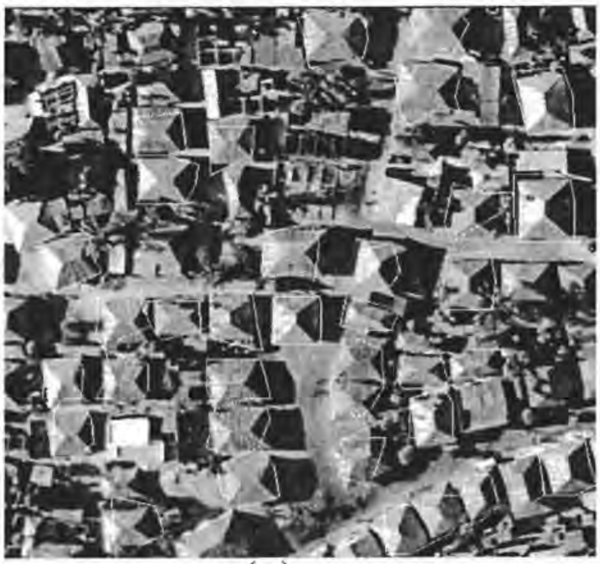
(a$_1$)



(a$_2$)

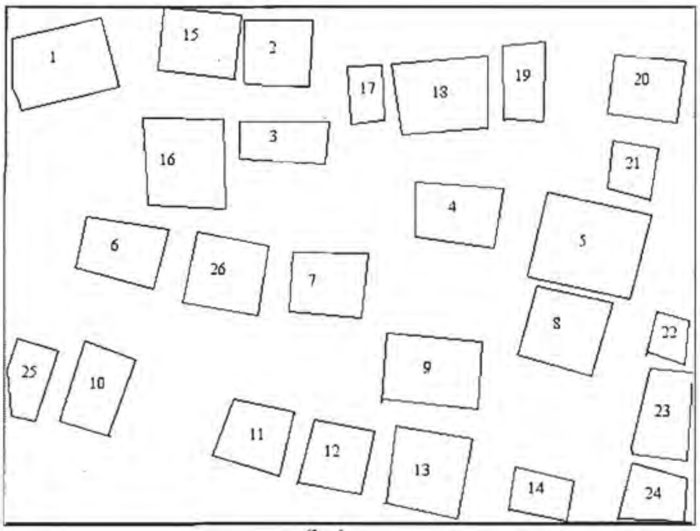


(a$_3$)



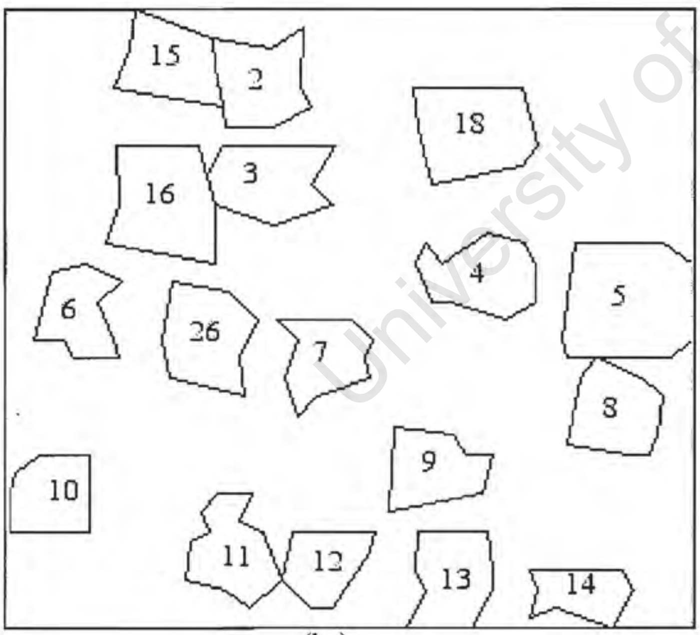
(a$_4$)

(a) Manzese Site1



(b₁)

(b₂)



(b₃)

(b₄)

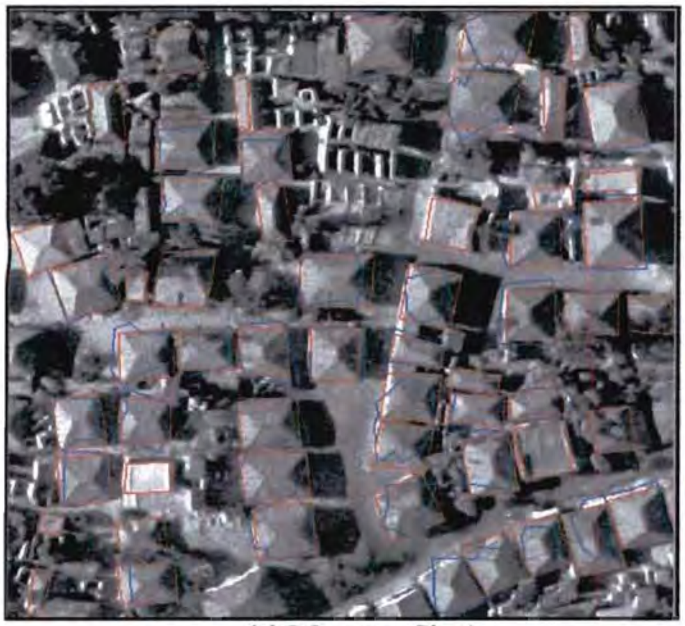
(b) Manzese Site2

(c1) (c2)

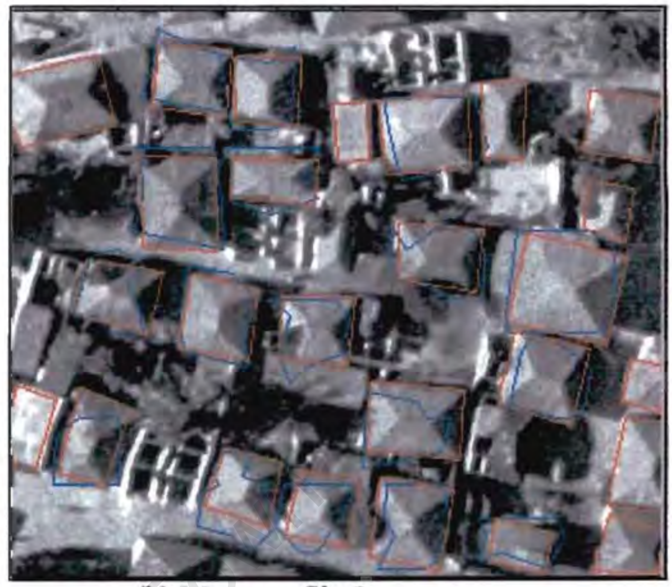
(c3) (c4)

© Marconi Beam Site

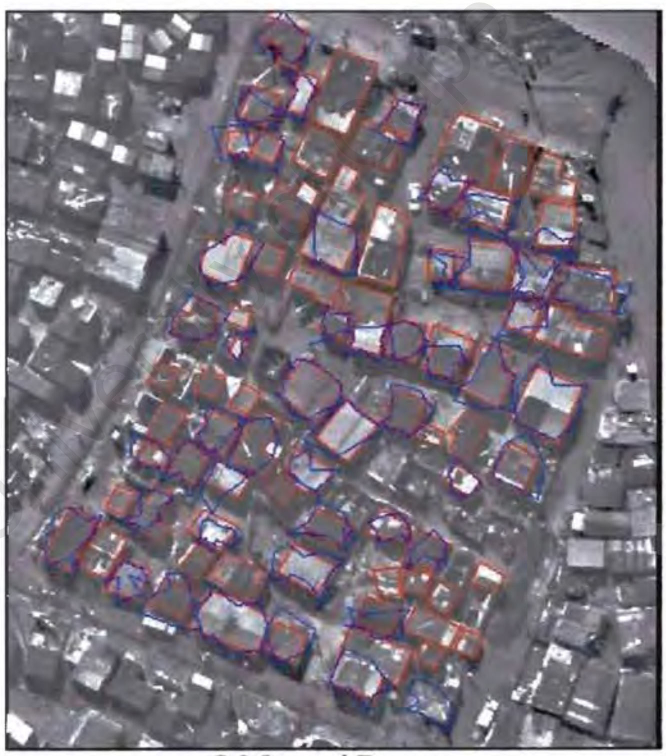
Figure 36. 2-D building extraction results along with ground truth data.
(a_1, b_1,c_1 – Ground Truth data; a_2,b_2,c_2 – Ground Truth data superimposed on orthoimages; a_3,b_3,c_3 – 2-D building extraction results; a_4,b_4,c_4 – Extracted buildings superimposed on orthoimages).



(a) Manzese Site1



(b) Manzese Site2



© Marconi Beam

Figure 37. Extracted buildings and ground truth data superimposed on orthoimages.

(Red – Ground Truth, Blue – Extracted Buildings).

5:2:4 PERFORMANCE EVALUATION OF THE PROPOSED BUILDING EXTRACTION SYSTEM

5:2:4:1 SYSTEM PERFORMANCE EVALUATION DESIGN CONSIDERATIONS

A fundamental requirement for the development of systems for automated cartographic feature extraction and an absolute pre-requisite for production applications is the rigorous evaluation of the system's performance. System evaluation requires the definition of a set of metrics i.e. measurable quantities relevant to user needs and meaningful in terms of expected system performance. McKeown, et al (2000) stated that, system's performance measures must be objective and quantitative, instead of scores based on an operator's judgements. The system's output must be measured against reference data from an independent method that is known to be of higher quality, which in most cases of automation means being manually generated. The use of different data sets in characterizing the systems' performance gives a better understanding of the true potential of the system and highlights possible weaknesses.

The designing of a feature extraction system's performance evaluation methodology depends on such factors as:

- Type of feature to be extracted – Each type of feature has its own characteristics, which need to be considered in evaluating its extraction, e.g. whether it is a 2-D or 3-D, whether location, geometry or attributions are of primary of interest.

- User requirements – Users may be interested in specific feature attributes.
- Extraction system's design architecture – Whether the system is semi-automated or fully automated. This may necessitate the evaluation of individual modules. Also as to whether the system is monocular, stereo or multi-image based.

In general, the adopted evaluation methodologies should as far as possible address the following three fundamental questions:

1. Is the extraction system's output reliable i.e. biased or not?
2. Given partially delineated buildings, does the extraction system detect and delineate the buildings correctly or not?
3. Is the inclusion of automated processes helpful to users or not? Does it reduce the user's workload and fatigue in the acceptable elapse time or not (Hsieh, 1996).

Buildings are among the most complex cartographic features to extract due to their wide variety of complex shapes, high probability of occlusions from surrounding objects and the complex scenes in which they usually occur. This complexity propagates into the systems designed to extract them, making the design of relevant and efficient metrics difficult (Mckeown and Shufelt, 1993; Mckeown, et al, 2000). A number of building extraction metrics are reported in the literature which characterize different aspects of the evaluation process or reflect the requirements of different user communities. For example, the building detection rate which indicates the presence of a building at a given location is one aspect of performance evaluation and building delineation which shows the boundaries of buildings is another aspect. Different metrics are required for each aspect.

5:2:4:2 APPLIED SYSTEM EVALUATION METHODOLOGY

The basis of the evaluation methodology was the comparison between the 2-D building scene models generated by the proposed extraction system with reference data. The reference data are the most accurate 2-D building models obtained manually from the ground truth data. The ground truth data for the Manzese area sites were obtained by manual digitization of an existing topographic map (a photogrammeric machine plot compiled from the very same aerial photographs used in this research) at a scale of 1:2500 (see Figures 36a_1, b_1 and c_1). For the Marconi Beam area ground truth data was manually created by an experienced operator by measuring building roofs in the orthoimage covering the area (see Figures 36c_1 and c_2).

Building extraction evaluation metrics in most cases are based on either an area/volume comparison or a building-by-building count. Area/volume metric methods are based on comparing the proposed extraction system's output label for each pixel or voxel with the reference data, and computing statistics based on the consistency of labels. On the other hand, if each building hypothesis produced by the proposed extraction system can be related to its corresponding building in the reference data, a measure of building detection can be based on the approach of computing the amount of overlap attained. This approach is also referred to as the building-by-building count metric method (Shufelt, 1996; Mckeown, et al, 2000). This method weights all buildings equally, but establishing correspondence may be difficult when multiple hypotheses overlap a single building, and when a number of buildings in the reference data set are connected (Shufelt, 1996; Mckeown, et al, 2000). To address the questions raised in § 5:2:4:1, the area and building-by-building count metrics were applied in this research.

In the area based metric method a simplified form to the approach adopted by Shufelt and Mckeown (1993) and Mckeown, et al (2000) has been applied. The simplification was necessary due to:

- The proposed building extraction system is at its infant stage of development, so it needs further improvement and innovations before it can be subjected to a precise or rigorous evaluation methodology.
- Scene and building complexity found in informal settlement areas. For example, delineating an isolated building is much easier than extracting the same building in a crowded urban area.

It is assumed that a building is considered detected if any part if its roof has been detected. Since there are two data sets in consideration i.e. from the proposed building extraction system which forms the building hypotheses and from the reference data, there are four possible categories for each building hypothesis:

- True Positive (TP): Both the extraction system's scene model and the reference scene model classify the hypothesis as being a building.
- True Negative (TN): Both the extraction system's scene model and the reference scene model classify the hypothesis as not being a building.
- False Positive (FP): The extraction system's scene model classifies the hypothesis as being a building, but the reference scene model does not.
- False Negative (FN): The extraction system's scene model classifies the hypothesis as not being a building, whereas the reference data scene model does.

To evaluate the proposed building extraction system's performance, the number of TP, TN, FP and FP are counted, and the following metrics are computed:

- Building Detection Rate Percentage = $(TP/(TP+FN))\times 100$
- False Alarm Rate = $(FP/(TP+FP))\times 100$

The building detection percentage is a simple metric, measuring the fraction of reference buildings that are correctly classified as buildings by the proposed extraction system. The metric is a measure of building detection performance and it gives an indication on whether the proposed extraction system is biased or not. The false alarm rate is a measure of the rate at which the system miss-detects buildings.

For the reasons stated earlier on in the case of the area based metric method, the building-by-building count metric technique was applied using a simplified approach to the method adopted by Mckeown and Shufelt (1993) and Mckeown, et al (2000). The adopted approach is based on the difference in area between each extracted building and the area of its corresponding reference building from the ground truth data. Let for example (A) be the area of a building in the ground truth data, and (B) be the corresponding area of the reconstructed building, then, the building-by-building metric is expressed by the overlap extent value, which is computed as:

Overlap Extent = $(1-(|A-B|/A)) \Rightarrow$ Success Rate.

This metric is applied in cases where there is overlap between the extracted building and its ground truth data (see Figure 38).

5:2:4:3 SYSTEM PERFORMANCE EVALUATION RESULTS

System performance was evaluated in both study areas. The results of area/volume metric based system evaluation are as shown in Table 1. It should be noted that because the proposed extraction system verifies building hypotheses prior to building extraction (refer § 4:7), values of TN and FP become meaningless in this case, and therefore the evaluation of the systems performance is only on the basis of the building detection rate.

	Manzese Site1	Manzese Site2	Marconi Beam
TP	34	17	48
FN	21	9	30
Detection Rate (%)	62	65	62

Table 1. Performance evaluation by the modified area/volume metric method.

Results of system performance by the modified building-by-building count method are as shown in Table 2.

No	Polygon ID	Ground Truth Area (m^2) (A)	Extracted Area (m^2) (B)	|A-B|	Success Rate (1-(|A-B|/A)) (%)
1	2	62	71	9	85
2	3	50	77	27	46
3	4	70	72	2	97
4	5	133	144	11	92
5	6	67	58	9	87
6	7	67	59	8	88
7	8	83	70	13	84
8	9	94	65	29	69
9	10	62	59	3	95
10	11	59	69	10	83
11	12	57	52	5	91
12	13	84	69	15	82
13	14	35	46	11	69
14	15	70	68	2	97
15	16	101	106	5	95
16	18	91	98	7	92
17	26	76	81	5	93
					Min = 46, Max = 97 **Mean = 85, σ = 13**

(a) Manzese Site2

1	1	106	125	19	82
2	2	109	83	26	76
3	3	123	119	4	97
4	4	107	138	30	72
5	5	93	82	11	88
6	6	83	68	15	82
7	7	120	166	46	62
8	8	128	102	26	80
9	9	95	61	34	64
10	10	79	116	37	53
11	13	88	95	7	92
12	14	95	58	37	61
13	16	161	127	34	79
14	17	82	64	18	78
15	18	74	80	6	92
16	19	93	101	8	91
17	20	93	80	13	86
18	21	103	106	3	97
19	22	80	72	8	90
20	24	86	70	16	81
21	25	97	112	15	85
22	26	101	58	43	57
23	27	103	103	0	100
24	28	80	51	29	64
25	29	81	145	64	21
26	30	74	88	14	81
27	34	96	102	6	94
28	38	88	88	0	100
29	44	53	66	13	75
30	45	49	65	16	67
31	48	91	106	15	84
32	51	87	81	6	93

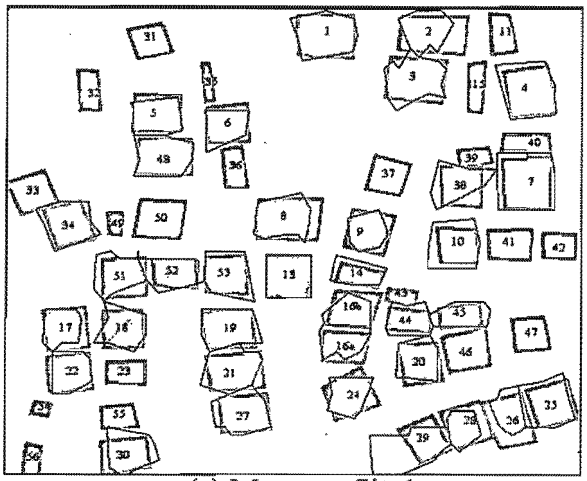
33	52	70	75	5	93
34	53	74	99	25	66
					Min =21, Max =100 Mean =79, σ=16

(b) Manzese Site1

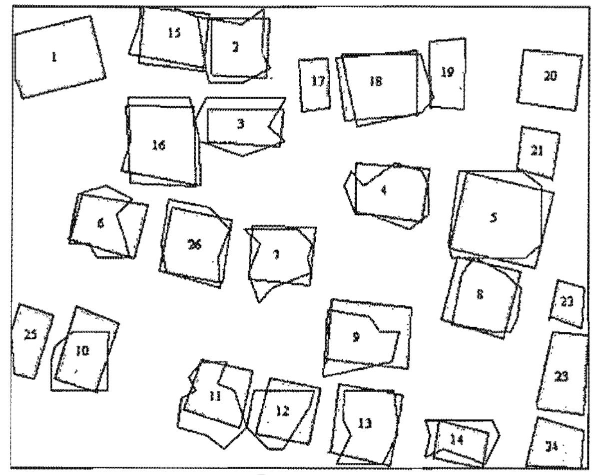
1	1	46	44	2	96
2	2	17	18	1	94
3	3	19	23	4	79
4	4	37	20	17	54
5	5	42	47	5	88
6	6	20	22	2	90
7	7	48	50	2	96
8	8	24	22	2	92
9	9	17	21	4	76
10	10	19	19	0	100
11	11	13	12	1	92
12	12	25	35	10	60
13	13	40	50	10	75
14	14	26	21	5	81
15	15	23	17	6	74
16	16	33	40	7	79
17	17	65	68	3	95
18	18	33	38	5	85
19	19	29	42	13	55
20	20	39	28	11	72
21	21	34	40	6	82
22	22	39	41	2	95
23	23	24	15	9	62
24	24	21	19	2	90
25	25	49	54	5	90
26	26	20	18	2	90
27	27	25	25	0	100
28	50	64	74	10	84
29	51	40	47	7	82
30	52	13	15	2	85
31	53	43	64	21	51
32	54	15	21	6	60
33	55	28	23	5	82
34	56	13	18	5	62
35	57	39	52	13	67
36	58	20	18	2	90
37	59	22	33	11	50
38	60	20	26	6	70
39	61	42	45	3	93
40	62	23	35	12	48
41	63	9	8	1	89
42	64	26	25	1	96
43	65	18	25	7	61
44	66	33	29	4	88
45	67	13	19	6	54
46	69	20	28	8	60
47	70	17	17	0	100
48	71	45	40	5	89
					Min =54, Max = 100 Mean =79, σ=15

(c) Marconi Beam

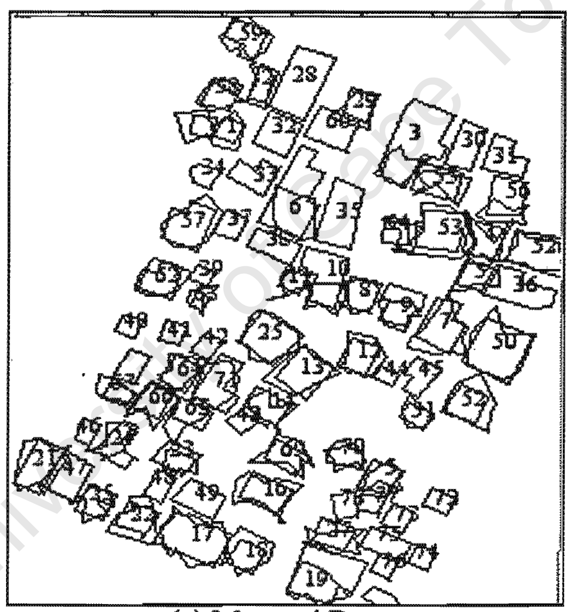
Table 2. Performance evaluation by the modified building-by-building count metric method.



(a) Manzese Site1



(b) Manzese Site2



(c) Marconi Beam

Figure 38. Extracted building contours superimposed on the ground truth data. (Shaded or filled areas – ground truth data; Polygons - extracted buildings). (NOTE: Most buildings in the ground truth data are extracted in each study area. There is a considerable amount of overlap between the extracted buildings and their corresponding buildings in the ground truth data)

5:3 3-D SITES MODELING AND VISUALIZATION

When considering a visualization system that is appropriate for a particular need there are several criteria that should be considered. These criteria go beyond the degree of interactivity, level of realism and presentation modes required to include portability across platforms and operating systems, user friendliness of the system, import/export capabilities with GIS data and other programs, links to models and data sets.

The ERSI's ArcView 3-D Analyst proved to fulfill most of the criteria mentioned above and was therefore applied. Additionally, the ArcView 3-D Analyst supports the generation of 3-D models in Virtual Reality Modeling Language (VRML). In ArcView 3-D Analyst the user is able to export 3-D models to the VRML exchange format. VRML browsers and plugins are currently inexpensive and widely available. By using VRML, created 3-D site models can be accessible via the Internet by hyper-linking it with the World Wide Web (WWW) where it would be available to a wide audience.

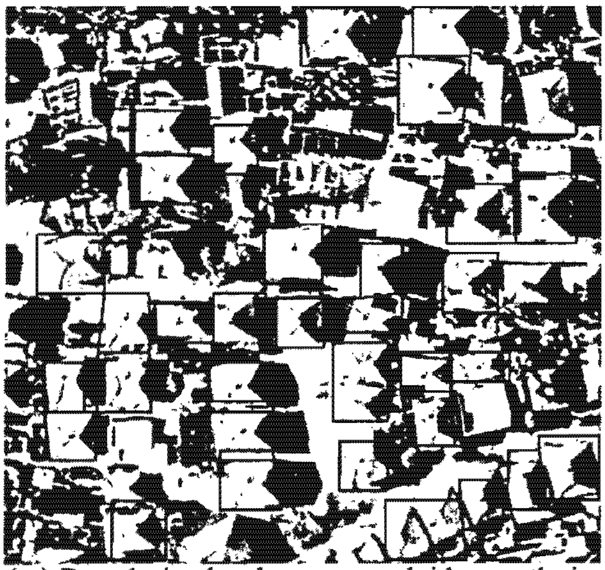
5:3:1 3-D MODELING BY ARCVIEW 3-D ANALYST

Inputs into 3-D Analyst are the DTM, orthoimages and the extracted 2D building contours. It should be noted that only the extracted buildings are modeled in 3-D merely to demonstrate the capabilities and completeness of the proposed method of building modeling. In a production environment, modules for the extraction of buildings that are not extracted by the proposed method need to be invoked in a separate process possibly a manual one prior to the 3-D modeling process. Because the final 2-D building outlines are rather distorted (see Figure 36a_3, b_3 and c_3) i.e. have irregular shapes, they are regularized by fitting a rectangular bounding box around each of them, by using an approach of minimum and maximum X and Y coordinate values of their vertices. This approach however did not work out well for the Marconi Beam study site, because extracted building contours were very irregular and very close to each other. This resulted in overlapping bounding boxes, the ARCVIEW 3-D Analyst failed to generate a 3-D model of the site. Dimensions and orientation of bounding boxes were regarded as dimensions and orientation of the buildings only for the purpose of visualization (see Figures 39a_1 and b_1).

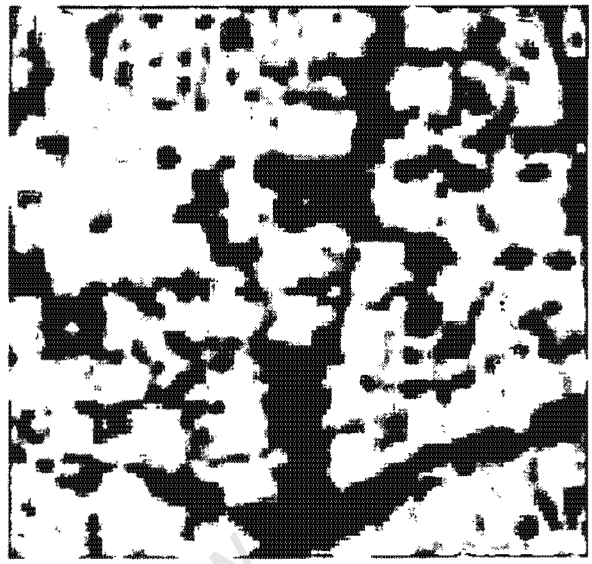
An approximate DTM obtained by gray scale erosion of the DSM using a kernel with a constant height value as used by Weidner and Förstner (1995) was applied as one of the inputs for the Manzese site (refer §4:6 for theory of gray scale erosion and see Figures 39a_2 and 39b_2 for approximate DTM's). In the implementation, the following procedures are applied:

1. The orthoimage is draped over a regular grid DTM forming a photo-realistic 3-D topographic surface in the ARCVIEW 3-D Analyst.
2. The extracted 2-D building outlines as approximated by rectangular bounding boxes are converted from Arc/Info polygon files into Arc/View shape files from which they form 2.5-D building blocks (i.e. blocks formed with correct planimetric positions but without actual height values).
3. Lastly, the 2.5-D building blocks and the photo-realistic topographic surface are merged together forming a photo-realistic 3-D site model.

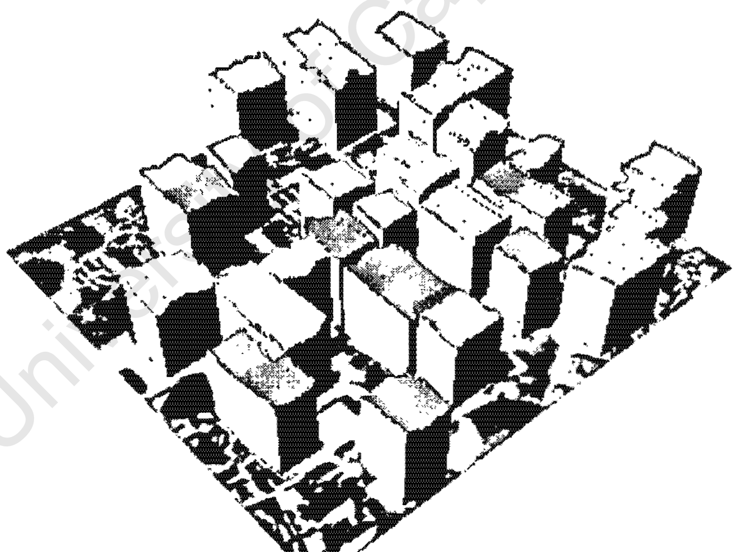
In the implementation, heights of building roofs theoretically are supposed to be extracted from the DSM, but in this research a fixed height value of 3m was assumed, as there were no significant building height differences. Under these circumstances, it is reasonable to assume a nominal height value for all buildings in the study area. The resulting 3-D site models (see Figure 39a_3 and b_3) open up opportunities for site visualization which may enhance user community participation in development planning.



(a₁) Regularized polygons overlaid on orthoimage



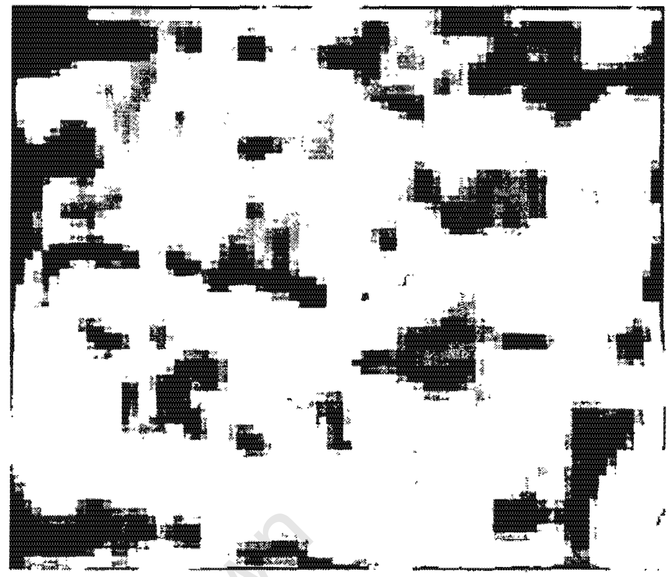
(a₂) Approximate DTM



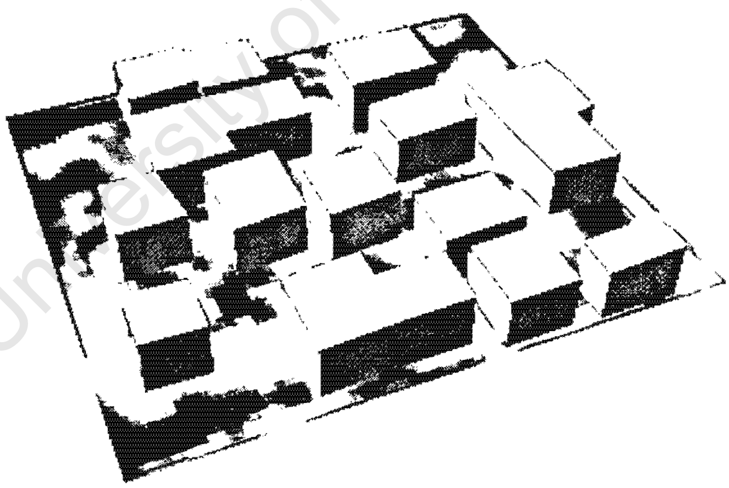
(a₃) Site Model

Manzese Site1

(b$_1$)Regularized polygons overlaid on orthoimage



(b$_2$) Approximate DTM



(b$_3$) Site Model

Manzese Site2

Figure 39. Regularized building contours, approximate DTM and site models.

CHAPTER 6

ANALYSIS OF RESULTS

This Chapter is analysing the results of DSMs altimetric segmentation, building hypothesis generation and verification and 2-D building extraction.

The results of DSMs segmentation by the global thresholding method as shown in Figure 18 in § 4:5 seem to adequately detect most of the building roofs in the Manzese study area. Limitations to the application of the DSM global thresholding method includes:

1. Difficulties in the selection of an effective global height threshold value which can detect most building roofs,
2. The effect of merged raised structure blobs and
3. The effect of the sloping terrain.

The success of the method depends on such factors as relative flat terrain and presence of adequate spacing in-between the buildings. In the case of small spacing in-between the buildings, the method fails as seen in the case of Marconi Beam site (refer Figures 16 and 17), where an alternative approach of Multiple Height Bins (MHB) was applied. It can be stated that digital surface models offer a reliable cue in raised structure detection and that the processing technique to be applied to raw DSM data should depend on circumstances surrounding the area of study. It should be noted that the accuracy of the surface models is vital for accurate raised structure detection.

In analysis of the process of building hypothesis generation from DSMs and the process of defining approximate building contours the following issues are highlighted:

1. Separation of merged raised structure blobs on the basis of shadows and binary erosion is not always effective as explained earlier on in § 4:6 and shown in Figure 19. One possible reason is that not all images are acquired under bright conditions with good illumination angles. As seen in Figure 19, a number of difficulties could arise e.g. the direction in which the raised structure blobs are merged may be different from that of the shadows. However, the approach of gray scale erosion of digital surface models has proved effective in this research. This suggests that one algorithm alone should not be expected to be effective in varying conditions found in the real world. Additionally this emphasizes the use of multi-algorithmic approaches in solving feature extraction problems.

2. Building hypothesis verification is necessary in distinguishing building hypotheses from other raised structure hypotheses. The method of verification needs to be as simple as possible to enable novice users in informal settlement settings to be able to do data processing without specialized technical training. The analysis of histograms of edge direction orthoimage patches that contain raised structure hypotheses as explained in § 4:7 has proved easier and more effective (see Figure 23) than computationally involved methods such as those based on the analysis of surface normal variances over the DSM raised structure blobs.

3. The method of region growing constrained by edges to generate approximate building regions has proved effective (refer Figure 24). The success of the method depends on such factors as the use of effective homogeneity criteria and the presence of contrast between roofs and surrounding features. In situations where the contrast is poor the method will definitely perform poorly. The output of region growing and edge detection need to be processed to remove artifacts prior to their use in the 2-D building extraction process.

From the 2-D building extraction results as shown in Figures 34 and 35, it can be stated that:

- The proposed building extraction method has been moderately successful with a success rate of above 60% (see Tables 1 and 2). It is expected that by detecting more than half of the buildings, the method reduces the amount of labour for the extraction task leaving undetected buildings to be extracted by other strategies. This implies the importance of multi-cue algorithms in feature extraction.
- It has not been possible to accurately delineate buildings with the same reliability and success rate as a human operator. Nevertheless, final building contours are closer to actual building lines as may be seen in Figure 36. However, they contain some artifacts which need to be removed by a manual post-editing step. The problem of inaccurate delineation of buildings may be attributed to:
  1. Differences in shape between approximate building contours and actual building contours.
  2. Adoption of a single (generic) building mathematical model for all buildings in a study area. For planned settlements it is logical to adopt a single generic building model, but the application of different building mathematical models

within a scene may be necessary for informal settlement areas. In other words, the adopted building mathematical model may not be adequately representing the manifestation of all buildings in the image.

3. Buildings being too close to each other. Automated delineation of very close buildings particularly from high resolution images is difficult because building lines tend to be ragged causing difficulty with their extraction. This stems from the fact that delineation of an isolated building is easier than extracting the same building in a complex and crowded area of informal settlements.

Comparing 2-D extraction results for the three study areas, it was found that buildings detected in the Marconi Beam site are relatively poorly delineated compared to those in the Manzese site (see Figures 36a_4, b_4 and c_4). This is because buildings in the Marconi Beam site are very close to each other, building sizes are very small and the buildings are made up of diverse building materials. Such circumstances lead to an erroneous approximation of building regions by the region growing and edge detection algorithms and consequently to poor building delineation. This is a limitation to the application of the proposed building extraction system.

Performance evaluation results of the proposed building extraction system on the basis of the modified building-by-building count metric method were satisfactory (refer Table 2). In the building-by-building count method, an overlap extent value of 100% will mean a very good building extraction performance. For the building-by-building method to work, it is logical to define an overlap extent threshold value above which a building is declared detected or not. Mckeown et al (2000) showed that an overlap extent threshold value of 50% is adequate to assume that a building is detected. Average overlap metric values of 85 and 79 have been obtained in this research (see Table 2). These results imply that on average each extracted building has greater than 80% overlap with its corresponding building in the ground truth data (see Figure 38). The performance of the proposed system is thus acceptable.

CHAPTER 7

DISCUSSIONS AND OUTLOOK

The purpose of this chapter is to discuss the applicability, and limitations of the proposed building extraction method.

A novel method of building detection and delineation from high-resolution aerial images has been presented. The proposed building extraction system may be implemented as a module on Digital Photogrammetric Workstations. If a digital surface model and an orthoimage are readily available, the method can be implemented on any computer system. To improve its efficiency, the algorithms can be integrated as a module within a GIS software package where processes like overlaying, geo-referencing, evaluation by comparison with ground truth and other data can easily be implemented because all data sets will be in one frame of reference. The important features of the proposed system are that:

- It does not rely on shadows in the detection and delineation of buildings as is usually the case in most existing building extraction systems.
- Approximate building contours are obtained automatically unlike other attempts where they are obtained manually by digitizing approximate corner points.
- It does not directly use a DTM in generation of raised structure blobs. This works under the assumption that the terrain is relatively flat. In the case of the sloping terrain a DTM will be required which may be generated by the technique of gray scale erosion of the digital surface model.

- Building contour modification is carried out by the snakes and dynamic programming optimization method. The method can work even in the presence of contrast gaps, fragmented edges and other distortions because of the use of global radiometric and geometric constraints.

All procedures of the proposed method are in principle automated. However, a human operator is needed to assist the system by carrying out such functions as setting thresholds e.g. the DSM segmentation global height threshold value and region growing homogeneity criterion values; assessing the visual quality of intermediate results and allowing the processing to continue or not, e.g. deciding if the results of region growing and edge detection are satisfactory or not; assessing if final results are acceptable or not and post editing of final building contours. Such use of a human operator within a system is considered minimal because visual assessments are easier and quicker for humans than computer systems.

Most of the buildings detected are simple rectangular shapes. Very few buildings in the study areas had complex figures of T, L, U or + shapes i.e. buildings with separate building segments or wings. Such building shapes are difficult to extract by the proposed method. This is because the method operates from centre points of DSM blobs to do region growing and edge detection in the determination of approximate building contours. In such complex building shapes the proposed system may not work well because it will be necessary for each building wing to have its own blob centre point. This is not possible in the proposed method, indeed, the possibility of joining separately extracted building wings into one has not been explored yet. It can therefore be stated that the proposed method is best suited for simple (preferably rectangular) shaped buildings of moderately complex and crowded informal settlement areas like the study areas used in this project.

The time efficiency for the proposed method has not been evaluated. This is because it is assumed that the method is expected to be run offline in which case time is not a major constraint. The method is however at its infancy stage and so it requires further innovations and improvements.

CHAPTER 8

CONCLUSION

8:1 SUMMARY

The proposed building extraction method has not been able to extract all the buildings in the study areas. This may be attributed mainly to scene and building complexity. Under different scene imagining conditions, the method is expected to perform differently. Development of a generic feature extraction system, which is likely to be robust enough to perform well in diverse scene conditions has been difficult to-date and is a challenge to research groups. This implies the need for using multi-cue algorithms in feature extraction. In multi-cue algorithms each extraction technique provides information that can be added or assimilated into an overall interpretation of the scene.

8:2 CONCLUSIONS

Images are usually acquired under different illumination conditions, time and viewing angles. Variations to any of these factors lead to corresponding changes in image qualities such as contrast, shadow effects and others which are leading cues in feature extraction. It is recommended that the method of building extraction be determined prior to the acquisition of images. It is logical to create structured images by controlling the imaging process. In other words, the extraction method should be considered as an issue when planning for the acquisition of images. By so doing acquired images may have the necessary cues for building extraction. For example, if an accurate and reliable DSM is not available it may be necessary to constrain the time of acquisition so as to get images with strong shadows that may aid in the building extraction process.

Although the main aim of the proposed building extraction method was to achieve full automation, it has been shown that some intervention by the human operator is still essential, thus leading to the development of a semi-automatic building extraction scheme. To this end it is reiterated that there is still a need for incorporating human interaction in building extraction systems due to the limited success rates of full automatic systems. Two types of interactions are required i.e. for on-line extraction processes and for the post-editing of derived results. The level of human intervention in the proposed system has been kept as minimal as possible, making the system automatic to a great extent.

It has been shown that the success of the snakes approach depends upon the initial state of the building contour, i.e. the approximate building contour needs to be as close to its true shape as possible. In the snakes approach, it is difficult to manipulate the approximate building contour to fit the expected contour if the difference in shape between the two is significant. In addition to that, the snakes approach suffers from computational complexity problems in dynamic systems like those with many degrees of freedom. In the light of the above, it is concluded that the snakes approach can be effective when applied after the human operator has manually determined the approximate corner points of a desired feature. This can be achieved by screen digitization of approximate corner points of the feature, which subsequently define the contour's initial state.

The proposed method is unable to correctly delineate building lines. This indicates that top-down strategies alone cannot effectively solve the problem of feature extraction. This is because, in these strategies, the search space becomes big and additionally elaborate object models are needed. One way to solve this problem could be by applying combined strategies i.e. combining top-down and bottom-up strategies. With combined strategies feature extraction can be carried out with continued reference to the image data and dynamically depending upon the local context knowledge of the objects. This seems to be the right way to approach the problem of feature extraction, as even the human vision system is based on integration of global and local context knowledge.

8:3 RECOMMENDATIONS

Owing to the limitations of the snakes approach, it is suggested that future work should look into the use of simulated annealing strategy in building extraction. Simulated annealing is analogous to the snakes approach with respect to energy optimization. It approximates the principle of annealing in physics where a crystal is cooled down from liquid to solid by the generation of sequences of states. The process can be simulated and applied in feature extraction (Trinder, Maulik and Bandyopadhyay, 2000). The method operates from the initial user delineated points which are modeled by B-splines, a rectangular window containing both the spline and the feature is defined and input into the simulated annealing process. The simulated annealing approach is reported to be effective in road extraction, and it has proved to be superior to the snakes approach in cases where the initial or approximate feature delineation does not closely approximate the feature of interest (Trinder, Maulik and Bandyopadhyay, 2000).

REFERENCES

Amini, A. A., T. E. Weymouth, and R. C. Jain, (1990), Using dynamic programming for solving variational problems in vision, *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence,* Vol.12 No.9, pp. 855-867

Anoroff, S., (1991), *Geographic Information Systems: A management perspective*. WDL publications, Ottawa.

Baltsavias, E, P., (1991), Multi photo geometrically constrained matching, *Ph.D. Thesis, No.49, Institute of Geodesy and Photogrammetry*, Zurich

Baltsavias, E., P., (1996), Digital ortho-image - A powerful tool for the extraction of spatial and geo-information. *ISPRS Journal of Photogrammetry and Remote Sensing*, 51, pp. 63-77

Baltsavias, E., S. Mason, and D. Stallman, (1995), Use of DTM's/DSM's and orthoimages to support building extraction, In *Automatic extraction of man made objects from aerial and space images,* Birkhauser Verlag, Berlin, pp. 199-210.

Ballard, D. H. and C. M. Brown, (1992), *Computer Vision*, Prentice Hall, New Jersey

Bellman, R. and S. E. Dreyfus, (1992) *Applied Dynamic Programming*, Princeton, New Jersey

Borrough, P.A., (1989), *Principles of Geographical Information Systems for Land Resource Assessment*, Clarendon Press, Oxford.

Brunn, A. and U. Weidner, (1998), Extracting buildings from digital surface models, *Conference on Computer Vision,* Thesolonike, Greece.

Canny, J., (1983) Finding Edges and Lines in images, *Technical report720, MIT, Artificial Laboratory,* Massachussets

Canny, J. (1986), A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAMI-8, pp. 679-698

Cho, K. and P. Meer, (1997), Image segmentation from consensus information, *Computer Vision and Image Understanding Vol. 68, No. 1*, pp. 72-89

Collins, R.T., C. O. Jaynes, Y. Cheng, X. Wang, F. Stolle, E.M. Riseman and A.R. Hanson, (1998), The Ascender system: Automated site modeling from multiple aerial images, *Computer Vision Graphics and Image Understanding*, Vol.72, No. 2 pp. 143-162.

Cooper, L. and D. Steinberg, (1970), *Introduction to methods of Optimization*, W.B. Saundres, Toronto.

Corr, D. G., A. M. Tailor, A. Cross, D. C. Hoggs, D. H. Lawrence, D. C. Mason and M. Petrou, (1989), Progress in automatic analysis of multi-temporal remotely sensed data, *International. Journal of Remote Sensing,* Vol.10, No.7, pp. 1175-1195.

Davies, E. R., (1990), *Machine Vision: Theory, Algorithms, Practicalities*. Academic Press, New York

Demetre, P. Argiallas and C.A. Harlow, (1990), Computational image interpretation models: An overview and perspective, *Photogrammetric Engineering and Remote Sensing,* Vol. 56, No.6, pp. 871-886

Erik de Man, (1988), Establishing a Geographic Information System in relation to its use. A process of strategic choices, *ITC Journal*, Vol. 2, No. 3, pp. 245-261.

Förstner, W. and E. Gülch, (1987), A fast operator for detection of distinct points, corner and centres of circular features. *ISPRS Intercommission Conference on fast processing of Photogrammetric data,* Interlaken, pp. 281-305

Fierens, F., P. Van Cleynenbreugel, P. Suetens and A. Oosterlinck, (1991), Iconic representation of visual data and models, *Pattern Recognition Letters, 12*, pp. 781-792.

Firestone, L., S. Rupert, J. Olson and W. Mueller, (1996) Automated Feature extraction: The key to future productivity, *Photogrammetric Engineering and Remote Sensing (June 1996),* pp. 671-674

Fua, P. (1998), Optimization techniques for feature extraction, *Computer Vision Conference,* Thesolonike, Greece

Fua, P. and T. Hanson (1987), Using generic geometric metric models for intelligent shape extraction, *Proceedings, DARPA Image Understanding workshop*, pp. 274-288.

Fu, K. S., (1982), *Syntactic Pattern Recognition and Applications*, Prentice Hall, New Jersey

Gairns, J. and T. Taylor, (1992) Feature Extraction comparison of image analysis system and Geographic Information Systems, *ISPRS, Commission IV*, Vol. XXIX, part B4, 365-367

Geo-information Africa (1998), issue 4, pp. 12-14.

Geo-Europe, (2000) issue 9, pp. 43-38.

Gorman, J. W., O. R. Mitchell and F. P. Kuhl, (1988), Partial shape recognition using dynamic programming, *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence,* Vol. 10, No.2, pp. 257-266.

Gorte, B., (1999), Tools for advanced image processing and GIS using ILWIS, *ITC Lecture Notes, Enschede,* The Netherlands

Gortschalk, P.G. and T. Mudge, (1998), Efficient encoding of local shape features for 3_D object recognition, *Intelligent Robots and Computer Vision, SPIE, Vol.1002,* pp. 42-57

Grenzdorffer, B. and R. Bill, (1994), Digital ortho photos for mapping and interpolation in hybrid GIS environments, *Fifth European Conference on Geographical Information Systems, pp. 1846-1855.*

Gülch, E., (1990a), Active contour models for semi automatic extraction of features in digital images. *International Archives of Photogrammetry and Remote Sensing*, Vol. 28 part3

Gülch, E., (1990b), Extraction of contours in digital images by active contour models, *International Archives of Photogrammetry and Remote Sensing*, Vol. 28 part3

Gülch, E., (1992), A knowledge based approach to reconstruct buildings in digital aerial imagery. *ISPRS, Commission II*, pp. 410-417

Gülch, E., (1995), Information extraction from digital images - A KTH approach, In *Automatic extraction of man made objects from aerial and space images*, Birkhauser Verlag, Berlin, pp. 73-82

Gülch, E., (2000), Digital systems for automated cartographic feature extraction, *International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, part3*, pp. 241-256

Gruen, A. and H. Li, (1995), Road extraction from aerial and satellite images by dynamic programming, *ISPRS Journal of Photogrammetry and Remote Sensing* Vol. 50, No.4, pp. 11-20.

Gruen, A. (1996), Development of digital methodology and systems, In Close Range Photogrammetry and Machine Vision, Edited by Atkinson, K.B. , Whitely, pp. 78-104.

Gruen, A. and H. Li, (1997), Semi automatic linear feature extraction by dynamic programming and LSB snakes, *Photogrammetric Engineering and Remote Sensing*, Vol. 63, No. 8, pp. 985-995.

Gruen, A. (2000), Semi automated approaches to site recording and modeling, *International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, part3*, pp. 309-318

Haala, N. and C. Brenner, (1997), Interpretation of urban surface models using 2D building information, In *Automatic extraction of man made objects from aerial and space images II*, Birkhauser Verlag, Berlin, pp. 213-222

Haala, N. and M. Hahn, (1995), Data fusion for the detection and reconstruction of buildings. In *Automatic extraction of man made objects from aerial and space images*, Birkhauser Verlag, Berlin, pp. 211-221

Haralick R. M. and L. G. Shapiro, 1992, *Computer Vision and Robot Vision, Vol.1* Addison-Wesley, New York.

Haralick R. M. and L. G. Shapiro, 1993, *Computer Vision and Robot Vision, Vol.2* Addison-Wesley, New York.

Haralick R. M., S.,R., Sternberg and X. Zhuang, (1987), Image analysis using mathematical morphology, *IEEE transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No.4

Henricsson, O., (1998), The role of color attributes and similarity grouping in 3D building reconstruction, *Computer Vision and Image Understanding* Vol. 72, No.2, pp. 163-184.

Henricsson, O. and E. Baltsavias, (1997), 3D building reconstruction with ARUBA: A qualitative and quantitative evaluation, In *Automatic extraction of man made objects from aerial and space images II,* Birkhauser Verlag, Berlin, pp. 65-76

Hsieh, Y., (1996), Site city: A semi automated site modeling system, In proceedings of *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 499-506 San Francisco, California,

Hoffman, R. and A. K. Jain, (1987), Segmentation and classification of range images, *IEEE transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No.5, pp. 608-620.

Hug, C., (1997), Extracting artificial surface objects from airborne laser scanner data, In *Automatic extraction of man made objects from aerial and space images II,* Birkhauser Verlag, Berlin, pp. 203-212

Huertas, A. and R. Nevatia, (1988), Detecting buildings in aerial images, *Computer Vision Graphics and Image Processing 41*, pp. 131-152

Ikonos, (1999), Space imaging global network, http://www.spaceimaging.com
(Accessed on 10/5/2000)

ISPRS Highlights, 1996, Vol.3 No.2, pp. 15-17

Jaynes, C.O., F. Stolle and R.T. Collins (1994), Task driven perceptual organization for extraction of roof top polygons, *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, pp. 152-159

Kass, M., A. Witkin, and D. Terzopoulous, (1988), Snakes: Active contour models., *International Journal of Computer Vision*, 1(4), pp. 321-331.

Kitchen, L., and A. Rosenfeld, (1982), Grey level corner detection, *Pattern Recognition Letters 1*; pp. 95-102

Koch, M. and R. L. Kashyap, (1987), Using polygons to recognize and locate partially occluded objects, *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, No.*4* pp. 152-159

Levine, M.D. and S.I. Shaheen, (1981), A modular computer vision system for picture segmentation and interpretation, *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, Vol. PAMI-3, No.5, pp. 540-556

Levitt, S. and F. Aghdasi, (1998), Texture measure for building recognition in aerial photographs, *Unpublished Msc. Report Wits University, South Africa*

Li, J. S. Mason and H. Rüther (1998), Fusion of multiple cues in object space to extract informal settlements in small format digital aerial images, *ISPRS Commission III symposium on object recognition and scene classification from multispectral and multisensor pixels.*

Liow, Y. and T. Pavlidis, (1990), Use of shadows for extracting buildings in aerial images, *Computer Vision Graphics and Image Processing 49*, pp. 242-277

Maas, H., (1999), Fast determination of parametric house models from dense airborne laser scanner data, *Proceedings-International Workshop on mobile mapping technology*, Bangkok, Thailand, 5A-1-1 to 5A-1-6.

Madani, M., (2000) Stand alone ortho rectification, GIM (International), March 2000, pp. 77-79

Mason, S. and E. Baltsavias, (1997), Image based reconstruction of informal settlements, In *Automatic extraction of man made objects from aerial and space images II,* Birkhauser Verlag, Berlin, pp. 97-108

Mason, S. O. and C. S. Fraser, (1998), Image sources for informal settlement management, *Photogrammetric Record* 16(92), pp. 313-330

Mason, S. and H. Rüther, (1997), Managing informal settlement spatially, presented paper: ASPRS Annual meeting, Seattle

Mason, S., H. Rüther, and J. Smit (1997), Investigation of the Kodak DCS 460 digital camera for small area mapping, *ISPRS Journal of Photogrammetry and Remote Sensing,* 52, pp. 202-214

Mather, P.M., (1987), *Computer Processing of Remotely Sensed Images. An Introduction,* John Wiley, New York.

Matsuyama, T., (1987), Knowledge based aerial image understanding systems and expert systems for image processing, *I.E.E.E. Transactions on Geoscience and Remote Sensing,* GE-25, pp. 305-316.

Matsuyama, T., (1989), Expert systems for image processing: Knowledge based composition of image analysis processes, *Computer Vision Graphics and Image Processing, 48*, pp. 22-49

Mckeown, D. M., (1990), *Towards automatic cartographic feature extraction, In Mapping and Spatial Modelling for Navigation,* Springer-Verlag, Berlin

Mckeown, D. M., (1984), Knowledge based aerial photo interpretation, *Photogrammetria,* vol. No.4, pp. 567-572

Mckeown, D. M., W.A. Harvey, J. Mcdermontt (1985), Rule based interpretation of aerial imagery, *I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAMI-7, No.5, pp. 570-585

Mckeown, D. M., T. M. Bulwinkle, S. Cochran, W. Harvey, C. Mcglone and A. Shufelt, (2000), Performance evaluation for automatic feature extraction, *International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, part3*, pp. 379-394

Monnier, V-P, P. Wong and B. Kolk, (1999), Digital ortho photos and facilities management, GIM (International), May 1999, pp. 44-47.

Moravec, H.P.,(1977), Towards automatic visual obstacle avoidance, *Proceedings of International Joint Conference, Artificial Intelligence*, pp. 584

Morgan, M. and K. Tempfli, (2000), Automatic building extraction from airborne lase scanning data, *International Archives of Photogrammetry and Remote Sensing,* Vol. XXXIII, part B3, pp. 616-623

Mulder, N.J., (1988), *Vegetation Mapping*. Kluwer Academic publishers, Dordrecht, Netherlands.

Mulder, N. J., (1990), Remote Sensing data as input for a GIS?,*URSA-NET*, Vol.7, Delft, Netherlands.

Msemakweli, C., K., (1999), Design considerations in the generation of digital terrain models by image matching for GIS databases, *Ph.D. Purdue University, USA*

Nagao, M. and Matsuyama, T., (1980), *A structural analysis of complex aerial photographs.* Plienum press, New York

Nazif, A. M. and M.D. Levine, (1984), Low level image segmentation: An expert system, *I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence,* Vol.PAMI-6, No.5, pp. 555-577

Nemhauser, G. L., (1966), *Introduction to Dynamic Programming,* John Wiley, New York

Nicolin, B, and R. Gabler (1987), A knowledge based system for the analysis of aerial images, *IEEE Trans. Geoscience-Remote Sensing* GE-25, pp. 317-329.

OEEPE, (2000), Project on laser data acquisition, http://www.geomatics.kth.se/~fotogram/OEEPE/wg_draft.htm, (Accessed on 30/4/2000)

Paparoditis, N., M. Cord, M. Jordan and J.-P. Cocquerez, (1998), Building detection and reconstruction from mid and high resolution aerial imagery, *Computer Vision and Image Understanding*, Vol.72, No.2, pp. 122-142.

Pavlidis, T. and Y. Liow (1990), Integrating region growing and edge detection, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol.12, No.3 pp. 225-233.

Pratts, W. K. (1991), *Digital Image Processing*, John Wiley, New York.

Richards, J. A., (1993), *Remote Sensing Digital Image Analysis, 2nd Ed.*, Springer Verlag, London.

Roux, M. and D. M. McKeown, 1994. Feature matching for building extraction from multiple views. *IUW*, Monterey, CA: pp. 331-349.

Saradjian, M. R. and J. Amini, (2000), Image map simplification using mathematical morphology, *International Archives of Photogrammetry and Remote Sensing,* Vol. XXXIII, part B3, pp. 36-43

Schenk, T., (2000), Object recognition in digital photogrammetry, *Photogrammetric Record,* 16(95), pp. 743-762

Seresht, M.S. and A. Aziz, (2000), Automatic building reconstruction from digital aerila images, *International Archives of Photogrammetry and Remote Sensing,* Vol. XXXIII, part B3, pp. 792-798

Sester, M., W. Schneider, D. Fritsch, (1996), Results on image understanding of ISPRS, *Working Group III/3, 18 th ISPRS Congress*, Wien.

Sheppard, S. R. J., (2000), Visualization software: Bringing GIS to life, Geo-Europe, Vol. No.9, Issue 8.

Shufelt, J., A. and D. M. McKeown, (1993), Fusion of monocular cues to detect man-made structures in aerial imagery, *Computer Vision Graphics and Image understanding*, Vol. 57, No. 3 pp. 307-330.

Shufelt, J., A., (1999), Performance evaluation and analysis of monocular building extraction from aerial imagery, *I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence, Vol.PAMI-21, No.4, Special Section on Empirical Evaluation of Computer Vision Algorithms*, pp. 311-326

Smit, J. L., (1997), Three dimensional measurements of textured surfaces using digital photogrammetric techniques, *Ph.D., University of Cape Town, South Africa.*

Smit, J. L. and Rüther, H., (1996), The 3D mapping of textured surface using digital photogrammetric techniques, *ISPRS, Vol. XXXI (B4),* pp. 728-733.

Suetens, P., P. Fua, and A.J. Hanson, (1992), Computational strategies for object recognition, *ACM Computing Surveys*, Vol. 24 , No:1, pp. 5-61.

Stilla. U., U. Soergel and K. Jaegar, (2000), Generation of 3D city models and their utilization in image sequences, *International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, part B2*, pp. 518-524

Tanimoto, S. L., (1987), *The Elements of Artificial Intelligence. An Introduction to LISP*, Computer science press, MaryLand, USA.

Trinder, J., Y. Wang, A.S.Sowmya and M. Palhang, (1997), Artificial intelligence in 3D feature extraction, In *Automatic extraction of man made objects from aerial and space images II,* Birkhauser Verlag, Berlin, pp. 257-266.

Trinder, J. C., U. Maulik and S. Bandyopadhyay, (2000), Semi-automated feature extraction using simulated annealing, *International Archives of Photogrammetry and Remote Sensing, Vol. XXXIII, part B3*, pp. 905-911

Trotter, C.M., (1991), Remotely sensed data as an information source for Geographical Information Systems in natural resource management: A review, *Int. Journal of Geographical Information Systems,* Vol. 5, No. 2, pp. 225-239

UNCHS, (1997) Programme on line. UNCHS (Habitat), *http:\\www.unchs.unon.org\uchs\indicat\indintro.htm.* (Accessed on 26/5/2000)

United Nations Development Programme (1992), Human Development Report, New York

Walker, A. S., (1997), Practical automation in commercial digital photogrammetry, *Photogrammetric Record*, 15 (89) pp. 657-664

Weidner, U., (1997), Practical surface models for building extraction. In *Automatic extraction of man made objects from aerial and space images II,* Birkhauser Verlag, Berlin, pp. 193-202.

Wang, Y., (1998), Digital Image matching with structural matching techniques. Full automatic image matching. *Geo-Information Magazine*, Vol.12. pp. 52-53

Weidner, U. and W. Förstner, (1995), Towards automatic building extraction from high-resolution digital elevation models, *International Journal for The Society for Photogrammetry and Remote Sensing,* 50(4), pp. 38-49

Wong, K.W. and W. H. Ho, (1986), Close range mapping with a solid state camera, *Photogrammety Engineering and Remote Sensing,* Vol.52, No.1 pp. 67-74

Zhang, (1992), Deformable model for image segmentation, *International Journal for The Society for Photogrammetry and Remote Sensing*

Zongjian, L., L. Jian and T. Chuang, (1990), Object oriented feature extraction from aero-imagery, *ISPRS Commission II WGII/III,* Wuhan China, pp. 184-187