Using an Anisotropic Diffusion Scale-Space for the Detection and Delineation of Shacks in Informal Settlement Imagery

Stephen Phillip Levitt

A thesis submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, Johannesburg, in fulfilment of the requirements for the degree of Doctor of Philosophy.

Johannesburg, October 2009

Declaration

I declare that this thesis is my own, unaided work, except where otherwise acknowledged. It is being submitted for the degree of Doctor of Philosophy in the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination in any other university.

Signed this ____ day of _____ 20__

_____

Stephen Phillip Levitt

Abstract

Informal settlements are a growing world-wide phenomenon. Up-to-date spatial information mapping settlements is essential for a variety of end-user applications from planning settlement upgrading to monitoring expansion and infill. One method of gathering this information is through the analysis of nadir-view aerial imagery and the automated or semi-automated extraction of individual shacks. The problem of shack detection and delineation in, particularly South African, informal settlements is a unique and difficult one. This is primarily due to the inhomogeneous appearance of shack roofs, which are constructed from a variety of disparate materials, and the density of shacks. Previous research has focused mostly on the use of height data in conjunction with optical images to perform automated or semi-automated shack extraction. In this thesis, a novel approach to automating shack extraction is presented and prototyped, in which the appearance of shack roofs is homogenised, facilitating their detection. The main features of this strategy are: construction of an anisotropic scale-space from a single source image and detection of hypotheses at multiple scales; simplification of hypotheses' boundaries through discrete curve evolution and regularisation of boundaries in accordance with an assumed shack model — a 4–6 sided, compact, rectilinear shape; selection of hypotheses competing across scales using fuzzy rules; grouping of hypotheses based on their support for one another, and localisation and re-regularisation of boundaries through the incorporation of image edges. The prototype's performance is evaluated in terms of standard metrics and is analysed for four different images, having three different sets of imaging conditions, and containing well over a hundred shacks. Detection rates in terms of building counts vary from 83% to 100% and, in terms of roof area coverage, from 55% to 84%. These results, each derived from a single source image, compare favourably with those of existing shack detection systems, especially automated ones which make use of richer source data. Integrating this scale-space approach with height data offers the promise of even better results.

To Giorgia,
for her unparalleled enthusiasm in trying to get her hands on this thesis.

Acknowledgements

I would like to thank my supervisor, Professor Barry Dwolatzky, for his guidance, encouragement and patience over the years. I would also like to acknowledge, Professor Ian Jandrell, for assisting me at a School level by allowing me the time and space to complete this PhD.

I would like to express my heartfelt gratitude to my wife, Alessia, and to the rest of my family for their love and support.

Finally, I would like to thank Group for sharing and easing the trials and tribulations of postgraduate study.

Contents

List of Figures

List of Tables

List of Algorithms

List of Symbols

The principal symbols used in this thesis are summarised below, and the first equation in which each symbol appears is given.

FP	False Positive, *Equation (2.1)*
TP	True Positive, *Equation (2.1)*
FN	False Negative, *Equation (2.1)*
$\partial_t u$	change in concentration u with respect to time t, *Equation (5.1)*
D	diffusivity tensor or function, *Equation (5.1)*
∇u	concentration gradient, *Equation (5.1)*
$I(x, y, t)$	image intensities of the smoothed 2D image at a particular scale t, and position (x, y), *Equation (5.2)*
$\frac{\partial}{\partial t} I(x, y, t)$	change in image intensities with respect to scale, *Equation (5.2)*
$\nabla I(x, y, t)$	image intensity gradient, and $\|\nabla I(x, y, t)\|$ = magnitude of the gradient, *Equation (5.2)*
$g(\,\cdot\,)$	diffusivity function, a function of the image gradient magnitude, *Equation (5.2)*
K	diffusivity constant, *Equation (5.3)*
$\Phi(x, y, t)$	flux function, *Equation (5.4)*
I_s^t	pixel intensity at position s in a discretely-sampled image, at time step or iteration, t, *Equation (5.5)*
λ	rate of diffusion, *Equation (5.5)*

η_s	spatial neighbourhood of pixel s, may be 4- or 8-connected, *Equation (5.5)*
$\|\eta_s\|$	number of neighbours of pixel s, may be 4 or 8 accordingly, *Equation (5.5)*
I_p	pixel intensity of a 4-connected neighbour of a pixel at position s, *Equation (5.6)*
$\nabla I_{s,p}$	difference between the intensities of neighbouring pixels at positions s and p, *Equation (5.6)*
H_s	homogeneous operator output at pixel position s, *Equation (5.7)*
$K(s_1, s_2)$	relevance of the vertex created by line segments s_1 and s_2, *Equation (5.8)*
$\beta(s_1, s_2)$	the turn angle in radians from s_1 to s_2, *Equation (5.8)*
$l(s)$	length of line segment s normalised with respect to the perimeter length, *Equation (5.8)*
S_{det}	samples classified as detections, *Equation (5.10)*
S_{non}	samples classified as non-detections, *Equation (5.10)*
S_{tot}	total number of sample points, *Equation (5.10)*
S_{type}	sample points of different types: *shadow*, *no-shadow*, *hypothesis*, *no-shadow-no-hypothesis*, *Equation (5.10)*
RS_{det}	number of roof-shadow line segments on which a detection occurs, *Equation (5.10)*
RS_{tot}	total number of roof-shadow line segments, *Equation (5.10)*
$\mathscr{P}_e(P)$	Euclidean perimeter of polygon P, *Equation (5.11)*
$\mathscr{P}_{cb}(P, \theta)$	city-block perimeter of P rotated by angle θ with the origin as the centre of rotation, *Equation (5.11)*
$\mathscr{R}(P)$	rectilinearity measure for polygon P, *Equation (5.11)*
$\mathscr{A}(P)$	area of polygon P, *Equation (5.12)*
$\mathscr{C}(P)$	compactness measure for polygon P, *Equation (5.12)*

C_G	number of permissible combinations for a group, *Equation (5.17)*
m	number of non-shadow-verified hypotheses in the group, *Equation (5.17)*
n	number of shadow-verified hypotheses in the group, *Equation (5.17)*
k	number of grouping candidates, *Equation (5.17)*
v_i	boundary of a particular shadow-verified hypothesis in the group being considered, *Equation (5.18)*
b	a grouped boundary formed by taking the convex hull of shadow-verified and possibly non-shadow-verified hypotheses, *Equation (5.18)*
O	objective function for expanded boundaries, *Equation (5.19)*
P_k	point support for point k, *Equation (5.19)*
l	length of an edge participating in an intersection, *Equation (5.20)*
t	length from the nearest endpoint of the edge to the intersection point, *Equation (5.20)*
A	area of a shack/building in the ground truth, *Equation (6.1)*
B	area of the corresponding extracted shack/building, *Equation (6.1)*

Nomenclature

2D	Two-dimensional
3D	Three-dimensional
ANOVA	Analysis of Variance
DEM	Digital Elevation Model
DCE	Discrete Curve Evolution
DMP	Differential Morphological Profile
DP	Detection Percentage
DSM	Digital Surface Model
DTM	Digital Terrain Model
FIS	Fuzzy Inference System
GIS	Geographic Information System
GSD	Ground Sample Distance
ISPRS	International Society for Photogrammetry and Remote Sensing
MDL	Minimum Description Length
MDS	Model-Driven Simplified
nDSM	Normalised Digital Surface Model
NDVI	Normalised Difference Vegetation Index
NF	Noise-Free
PCA	Principal Component Analysis
QP	Quality Percentage

RAM Random Access Memory

RGB Red, Green and Blue

ROC Receiver Operator Characteristic

Chapter 1

Introduction

Informal settlements as defined in [1] are dense settlements which are characterised by small makeshift shelters, a lack of formal planning, and other factors. Such settlements are not formally planned as they are constructed without the approval of municipal authorities and have a quasi-legal or outright illegal status. Consequently, residents lack basic infrastructure and services in addition to land tenure rights. Mason & Fraser [1, p. 313] state that these settlements are "a common feature of developing countries and are typically the product of an urgent need for shelter by the urban poor."

Informal settlements are a burgeoning phenomenon. According to a United Nations Centre for Human Settlements report published in 2003, nearly one billion people are currently "slum dwellers", which is broadly defined and includes informal settlements, and this number is likely to double in the next thirty years [2]. On the African continent much of the growth being experienced by cities is due to expanding informal settlement populations and it is predicted that these settlements will shelter 50% of the total urban population over the next two decades [3]. In South Africa close to 20% of the population is thought to be living in informal conditions [1] and trends indicate that informal settlements in or close to major cities will continue to grow in the future. Longitudinal studies of settlements in and around Cape Town show that the number of shacks is increasing at a rate of 10% per year [3].

The growth of such settlements, with adverse social, economic and environmental impacts, poses a significant challenge for the developing world in terms of sustainable development. Public policy regarding informal settlements has changed over the last few decades [4] and today the management and *in situ* upgrading of settlements is seen as a viable solution to the housing problem [3, 1].

In order to effectively manage settlements up-to-date spatial information is required and forms the basis for applications dedicated to:

- settlement detection in urban areas,
- digital terrain model (DTM) generation for environmental risk analysis,
- monitoring settlement expansion and infill,
- counting shacks for social surveys and determining electoral boundaries,
- urban planning, including the upgrading of electrical and telecommunication infrastructure, as well as housing, roads and water and sewerage systems [1, 5, 6].

Abbott & Douglas [3] note that the use of imagery to analyse informal settlements is well established and more than a decade old. The resolution of the imagery varies but for tasks such as urban planning relatively high-resolution images are required as it is necessary to map individual shacks. These raster images typically form the base data for spatially analysing settlements using a geographical information system (GIS) [7]. Two-dimensional (2D) vector maps which use polygons to delineate shacks and other features are derived from the raster images.

For scenarios in which individual shacks need to be mapped, nadir-view aerial images are most commonly used as they are capable of delivering the minimum required ground pixel resolution of 0.5 m [6]. However, it is worth noting that satellites are increasingly being used in imaging settlements and although their resolution continues to improve, it is not yet sufficient for the automated detection of individual shacks [8, 9]. Interestingly, some innovative, very low-cost imaging solutions are appearing. For example, it is possible to produce rectified and geo-referenced aerial photographs through attaching a digital camera to a plastic balloon inflated with hydrogen gas [10]. These images can currently be produced with a ground sample distance (GSD) of up to 0.4 m.

Informal settlements undergo rapid change when compared to formal settlements. For example, hundreds of new shacks can be built within a few months [11]. Additionally, the shack structures themselves may be frequently rebuilt and altered. As a direct consequence of this, Abbott & Douglas [3, p. 16] highlight the importance of longitudinal studies:

> "Time-series spatial analysis ... should be used for policy formulation. It provides sufficient information to identify trends, and thus determine the current, as well as probable future magnitude of settlements relative to the

> city as a whole. Thus, for example, when coupled with data on housing provision and finance, it lays the basis for a realistic, multipronged shelter strategy for a city. Similarly, it can be used for infrastructure planning or the assessment of social services."

The dynamic nature of these settlements requires that they are frequently imaged and mapped in order to obtain up-to-date and relevant base data. In order to effectively produce base map data the technologies used must be "low cost, both in data acquisition and information extraction, fast and reliable, simple to use, and as far as possible, based on off-the-shelf technologies." [5, p. 435]. This view is echoed by Mason & Fraser [1, p. 315]: "Hence the desire for mapping systems which are both very rapid and amenable to some level of automation, while at the same time being low cost."

This thesis focuses on the "information extraction" or "automation" aspect, particularly with respect to automating the detection and delineation of shack boundaries. Previous work has shown that partially automating shack detection produces a significant decrease in the amount of time taken to interpret images [12, 13]. Automation reduces both the cost and effort of interpreting images in addition to increasing the speed at which base mapping data can be produced. In large datasets, and cases where settlements are imaged multiple times, the gains to be had from automating the detection of shacks as far as possible will be substantial.

Automated shack detection is a difficult problem, which is evident when reviewing the literature. Few *automated* systems exist and those that do require height data and often have difficulty in detecting shacks and/or accurately delineating shack boundaries. This can be contrasted with a fair number of automated systems that exist for identifying buildings in formal settings from a single image, without relying on height data (see [14] for an overview of some of these). An alternative approach that is used to make the problem more tractable is to create *semi-automated* systems in which a human operator is responsible for the initial identification of the shack and the system then attempts to delineate the boundaries.

Some of the difficulties that arise when dealing with informal settlements images are that:

- Individual shack roofs are often constructed from a variety of disparate materials and therefore the roofs do not appear as homogeneous surfaces. This compounds the detection problem significantly when using image data by making it harder

to discern the roof boundary as opposed to the boundaries of the materials which compose the roof or of objects lying upon the roof.

- Shacks do not exhibit the same kind of geometrical regularity that can be expected from formal buildings. Geometrical regularity is therefore less useful as a detection cue in informal settlement scenes.

These two factors make it hard to implicitly or explicitly formulate descriptions of the objects being detected.

In addition to the above difficulties:

- Shacks tend to be closely clustered which affects the reliability of both shadow cues and height data. For closely spaced shacks it may not be possible to separate their shadow and height information, which results in aggregated structures being detected. Additionally, shadows may be fully or partially occluded if they fall on adjacent shack walls.
- Informal settlements may contain formal buildings. A successful detection strategy needs to be flexible enough to identify both shacks as well as buildings.

On the other hand there are mitigating factors:

- Useful end-user applications can be developed which do not require high precision shack extractions. For example, an approximate shack boundary or centre point is acceptable for house counts, building density measurements and electrical reticulation planning [15]. For many informal settlement applications positional and object modelling accuracy is of less importance than the need for spatial data which is frequently updated [6].
- Shacks generally have a relatively simple building and roof architecture that allows them to be simply modelled and potentially results in a less complex detection strategy.
- Informal settlements are often characterised by a lack of vegetation [6] which implies that the shacks being detected will generally not be occluded. This is helpful as partially visible structures complicate the detection task.

In this thesis a novel approach for automating the detection and delineation of individual shack roofs is presented which attempts to address the problem of non-uniform roof appearance. This difficulty is tackled by radically transforming the source image in such a way so as to homogenise the appearance of shack roofs. The image transformation is based on the use of an anisotropic scale-space which

produces a family of images, each of which is a blurred version of the original. A detection strategy is then proposed which, unlike existing systems, performs shack detection at multiple scales rather than at a single scale. This strategy is region-based which demands careful consideration of how hypothesis boundaries are to be both regularised and localised in the absence of strong geometrical constraints, whilst not ignoring the fact that shack boundaries tend to be rectilinear.

The strategy is validated through the construction of a prototype system which is applied to a number of different source images and is capable of delineating shacks with a reasonable degree of success. Standard metrics are used in evaluating the system's performance and in comparing it to systems which do not utilise a scale-space approach.

A complete system for automating the detection of shacks would involve a number of different components[1] including:

- A photogrammetric component for geo-referencing and rectifying source images and performing other photogrammetric operations.
- A computer vision or image analysis component for performing the detection and delineation of the target objects.
- A GIS component for spatial data presentation, annotation, processing and editing.

In this thesis the focus is on the computer vision/image analysis component and the prototype that has been constructed is a stand-alone MATLAB application devoted to this aspect, although, in future it may be advantageous to implement the computer vision aspects within a GIS system, as advocated in [13]. A GIS system would offer direct support for overlays, vector map editing and other benefits.

This thesis is structured as follows:

Chapter 2: This chapter provides a general background to the problem of automating building detection from aerial images. The main concepts and terms which describe this particular area of computer vision are presented and explained. These include the different domains of "image", "scene" and "world" and how they are related, as well as the different levels of representation within a vision system. Three key facets of building detection systems are discussed: the source data, the various models involved, and the types of extraction strategies. This chapter concludes by

[1] These components are identified in [13] but used in a slightly different sense here.

considering how the goals of building detection systems differ and how such systems are evaluated.

Chapter 3: Existing building detection systems and approaches which are relevant to the work at hand are reviewed. These include systems which utilise single images as source data, systems which make use of a scale-space approach, and systems dedicated to the detection of shacks from informal settlements. Finally, existing techniques for region boundary localisation and regularisation are considered as this is a key aspect of the detection strategy adopted.

Chapter 4: The shack detection problem is presented and defined as tackled in this work. An examination of the difficulties posed by roof substructure, which is particularly prevalent in informal settlement imagery, is given. The manner in which existing systems deal with these difficulties is reviewed. The use of a scale-space approach as a viable alternative to dealing with substructure issues is motivated and the potential benefits are discussed.

Chapter 5: The detection strategy that has been formulated is described in detail, stage by stage. The chapter begins by discussing the construction of an anisotropic scale-space. This is followed by the details of how homogeneous regions, which are viewed as shack hypotheses, are extracted at all scales and their boundaries simplified in order for shape analysis to take place. The method for hypothesis verification through the use of shadow is given, followed by a discussion on how hypotheses' boundaries are regularised in accordance with the assumed object model. The fuzzy rule system for selecting competing hypotheses from different scales is presented. Descriptions of the final two stages of the detection strategy — the grouping of hypotheses and localisation of their boundaries based on edge information — round out the chapter. Intermediate results from the different detection stages are presented throughout this chapter, using one of the test images.

Chapter 6: The results of applying the detection strategy described in Chapter 5 to a number of different source images are given here. Performance is assessed both qualitatively and quantitatively. For the quantitative assessment, both detection and quality percentages are determined for each of the source images. In addition to this, a more in-depth analysis is provided, in which consideration is given to how these metrics vary per detection stage and per boundary type. Shape accuracy metrics are also provided. Importantly, the contribution of each stack level to the final interpretation of the scene is assessed. The above results are analysed for each of the images and a comparison, where possible, with existing systems is given. The

remainder of the chapter is devoted to an evaluation of the execution time of each of the system's detection stages and a discussion of how sensitive the results are to the variations in the system's parameters.

Chapter 7: The findings of this thesis are summarised here, as well as the specific research contributions that have been made. Avenues for future research are suggested.

Auxiliary material is provided in the following appendices:

Appendix A: This appendix provides the complete and detailed results for all of the test images excepting the "walkthrough" image whose detailed results are given in Chapter 6.

Appendix B: The parameter values that have been used in the different detection algorithms are detailed here. These parameters are image invariant.

Appendix C: The prototype's fuzzy rule/inference system has been implemented using MATLAB's Fuzzy Logic Toolbox. This appendix presents the details of how the fuzzy membership functions are constructed and the 'FIS' file which MATLAB uses to represent the fuzzy system.

To aid the reader, each chapter begins with a summary of the main points covered and ends with a brief introduction to the following chapter.

The following chapter provides a general background to building detection in aerial images. This forms a solid basis for understanding Chapter 3, in which specific detection systems are reviewed.

Chapter 2

Background to Building Detection

In this chapter concepts and issues relevant to the task of building extraction from aerial imagery are discussed. The broad computer vision concepts of image, scene and world domains, as well as the different abstraction levels involved in image interpretation are described. This is followed by an examination of the key aspects involved in the building detection scenario, namely, the source data, the various models that are employed, and the detection strategy itself. The different end-user applications for which building detection systems are designed, and how this affects their goals, is addressed. Lastly, standard metrics for evaluating the system performance are presented.

2.1 Introduction

Image interpretation techniques have evolved from general methods for image processing and classification, which are applicable in many spheres, to techniques which require extensive and specific knowledge about a narrow domain and are used to create very specialised applications [16, 17]. One such domain is "aerial image understanding". Aerial image understanding describes a fairly broad category of vision systems that attempt to automatically or semi-automatically identify objects or entities within aerial photographs. Aerial image understanding systems, themselves, have widely varying goals from identifying individual objects, such as houses and trees, to land-use classification. This chapter presents the background to systems which have as a primary goal the detection or extraction of buildings. The term "building" is used here in the general sense and includes formal buildings, industrial buildings, shacks and so on.

Note that there are efforts to construct image understanding systems that represent their knowledge in a manner that is generalisable to domains other than that for which they were originally designed [18]. However, it is evident that most of the building extraction systems developed thus far (see chapter 3) are purpose-built systems with implicit knowledge [17].

Building detection is an active research field enjoying a large number of publications, especially in recent years. Three important collections of papers, entitled, *Automatic Extraction of Man–made Objects from Aerial and Space Images (I), (II)* and *(III)* were published in 1995, 1997 and 2001 respectively [19, 20, 21]. A special issue of the journal *Computer Vision and Image Understanding* dedicated to "Automatic Building Extraction from Aerial Images" was published in 1998 [22]. There are also a number of journals concerned with photogrammetry and remote sensing that contain articles related to building detection. A key journal in this area is the ISPRS Journal of Photogrammetry and Remote Sensing, published by the International Society for Photogrammetry and Remote Sensing [23]. This society promotes research into many aspects associated with deriving information from imagery and hosts regular symposiums and congresses. Finally, a few survey papers have been published some of which focus directly on building detection, while others deal more broadly with aerial image understanding [17, 24, 25, 26, 16].

A building detection system involves the following key aspects:

- the *source data* which forms the input to the system;
- the *strategy* in combination with various *models* that is employed for detecting or extracting buildings;
- the output of the system which is determined by the *system goals*.

These aspects are used as criteria for assessing different building extraction approaches [17]. They also provide a useful framework for understanding existing systems. Subsequent sections of this chapter explore this framework in more detail and highlight issues which are relevant to the detection of buildings from aerial images. A review of the specifics of existing systems is left to the following chapter, chapter 3.

2.2 Image Domain, Scene Domain and World Domain

Three domains or levels of abstraction that can be used to describe the information contained in an image are defined [16]. The *image domain* corresponds to the lowest

level of abstraction and consists only of spatially varying pixel intensities which form the digital image. These are obtained via the imaging geometry of the light source, object and sensor. The image domain is one of incontrovertible fact.

The *scene domain* requires an interpretation of an image or the image domain. This domain is abstract and consists of semantic or meaningful objects and their relationships to each other. For instance, in an aerial image of suburbia, the pixel intensities of the image domain would correspond to "house", "road" and "swimming pool" objects in the scene domain. To move from the image domain to the scene domain is a large component of computer vision, and it requires both knowledge and intelligence. Knowledge about the objects being searched for must either be modelled implicitly or explicitly. An example of such a model would be an expected configuration of edges. A large cognitive gap exists between the image domain and the scene domain. It is, therefore, necessary to introduce intermediate levels of representation when moving from the former domain to the latter. These levels are described in section 2.3.

The *world domain* consists of describing all physical objects in three-dimensional space. The world domain description is thus no longer viewer-centric.

2.3 High-, Mid- and Low-Levels of Representation and Associated Processes

A key input to many building detection systems discussed in this thesis is source imagery composed of pixels of varying intensity. The output of such systems are two or three-dimensional models of the buildings that have been found. The large gap that exists between the input image and the output interpretation is bridged via a range of representations [25]. Different processes are used to move from very low-level representations, which are close to the image domain, to high-level representations of the scene. Three levels of representing the information contained in an image are usually recognised, corresponding to different levels of abstraction — high, medium and low [27].

High-level representations describe the image in terms of highly semantic constructs. A high-level representation of a scene allows the scene to be interpreted in terms of the final goals of the analysis [16]. Although high-level objects are the ultimate goal of a vision system, these objects have to be synthesised from mid-level objects, which, in turn, are constructed from the aggregation of primitives or low-level objects.

Primitives do not have any semantic content (they are domain independent) and form the most basic elements in the representation hierarchy.

Different processes are employed depending on the level of representation that is being dealt with. Examples of low-level algorithms include edge and arc detection, thresholding, and spatial and statistical classification. Mid-level processes include feature extraction, region growing, curve linking, grouping and relaxation labelling, amongst others. These processes are distinguished by the fact that they create a structural description from the raw pixel data but incorporate little or no domain-specific knowledge. Building detection systems often utilise geometric models, and hence rely heavily on *grouping* mid-level processes which organise primitives such as edges according to geometric regularities [28, 29].

High-level processes explicitly represent and use knowledge concerning the class of scenes that are to be understood. They are more sensitive to the context of the scene and more responsive to specific recognition needs [30, 27]. Explicit knowledge representation brings with it many benefits [18] and has taken a variety of forms in aerial image understanding systems, including logic-based representations, rule-based and production systems, semantic networks, blackboard systems, and frames [25, 24]. Often the emphasis of systems which utilise high-level processes is the interpretation of entire aerial scenes as opposed to the detection of specific target objects, as in [31]. High-level processes are employed to integrate diverse object detection modules. These processes are also used to deal with complex object models. According to Baltsavias [24] the choice of a knowledge representation framework is not a crucial issue as long as the underlying object extraction modules are sound and the framework which is chosen is sensible.

Many systems, however, do not explicitly represent the knowledge in models [17] and consequently lack the high-level abstraction layer. Instead, the knowledge is said to be "implicit" because it is indistinct from the detection strategy and is not formally represented. In such systems, a strong declarative description of the buildings being searched for does not exist. Buildings are implicitly defined by the algorithms used to detect them. This results in more efficient systems at the cost of generality [18].

2.4 Source Data

The source data which forms the input to a building detection system largely influences the extraction strategy employed, the object models that are used, and ultimately,

the complexity of the entire system.

There are two main aspects to consider with regard to the source data, namely, scene content and object observability [17].

2.4.1 Scene Content

Factors affecting scene content include [17, 32]:

Structure Density

Rural scenes generally exhibit low density housing; suburban scenes have medium density housing; urban areas tend to have a high density of man-made structures. Informal settlement areas are typically crowded with a high structure density. For lower structure densities a successful interpretation is more attainable, assuming that all other aspects are equal. This is due to the fact that building structures that stand alone can be more easily distinguished from background scenery and are less likely to occlude neighbouring structures. Additionally, supporting evidence for each structure, such as shadow, is less likely to be disturbed if there is sufficient distance between structures.

Object Complexity and Architecture

Building structures may differ greatly in their complexity and architecture. Ground plans may vary from circles (rondavels) to rectangles (shacks and A-frame houses) to more complex rectilinear and non-rectilinear shapes typical of military and industrial buildings. Roof architecture may vary from being simple (flat-roofed buildings) to highly complex (multi-plane roofs with gables and dormer windows). The three-dimensional shape of a structure is important as it affects its two dimensional appearance in an image. For example, consider a flat-roofed rectangular structure and a simple A-frame house with identical and uniform roofing material. Though both structures have rectangular ground plans, their appearance in an aerial image may be quite different. The flat-roofed structure may appear as a single rectangle of homogeneous colour while the two planes of the A-frame's roof may appear significantly different in colour if one plane is sun facing whilst the other is not, potentially making it more difficult to extract.

Terrain and Vegetation

The terrain may be flat or hilly, or even mountainous. Uneven terrain can result in perspective distortion which makes the recognition task more complex.

Such distortion can be corrected through ortho-rectification of the images if ground relief data is available. The greater the amount of vegetation in a scene, the more likelihood there is of building structures being occluded. For example, trees which overshadow buildings disturb the building outlines and complicate the recognition task.

2.4.2 Object Observability

Factors affecting object observability are [17]:

Resolution

The most fundamental factor related to object observability is that of resolution. Mayer [17, p. 140] notes that "analogously to the Nyquist theorem, an object has to be sampled with a spatial resolution which is half the size of the object to be distinguished from other objects." If the system goal is to clearly delineate the object boundaries or to identify substructure (like gutters) then a high resolution is required. Resolution is directly dependant on the scale of the aerial imagery (scales typically vary between 1:70 000 and 1:4 000) and on the scanning resolution if digital imagery is produced by scanning analogue film [17]. For example, scanning a 1:30 000 image with a resolution of 15 µm results in a ground pixel size of 45 cm.

The commercial use of digital aerial cameras is increasing and these cameras offer the advantages of a completely digital workflow, better radiometric image quality and the option of simultaneously acquiring panchromatic, colour and near-infrared imagery [33]. Additionally, they represent a comparatively low cost solution to analogue film in situations where the area in question is relatively small and needs to be frequently imaged, such as informal settlements [1]. Small format digital cameras can produce a ground resolution of around 0.18 m at a flying height of 520 m. Higher resolutions are possible but are deemed to be impractical [1].

Three resolution categories for aerial photography have been identified by Mayer:

1. Low: ground pixel sizes > 1 m
2. Medium: ground pixel sizes ≥ 0.2 m and ≤ 1 m
3. High: ground pixel sizes < 0.2 m

Decreasing the resolution of the source imagery has been shown to negatively

impact on the performance of building detection systems [34, 29, 35]. For example, in [34] the extraction strategy described in [36] is applied to source images, with varying ground pixel sizes, of the same urban area. As the ground pixel size increases the number of correctly identified buildings decreases. Difficulties with lower resolution images arise due to the following:

- The power to resolve individual objects diminishes. At lower spatial resolutions an object will be represented by fewer pixels in the image domain. Additionally, object proximity increases in the image domain as fewer pixels separate neighbouring objects. These factors make it difficult to distinguish a single object from adjacent objects.
- Low-level geometric primitives, such as edges, become difficult to extract in the presence of a reduced signal-to-noise ratio. For example, the difference in edge length between genuine edges and spurious edges may become negligible in low resolution imagery, making it hard to discriminate between the two. If the extraction strategy is dependent on edges then it may lack the minimum density of edge cues required for follow-on grouping processes.

It has also been shown that strategies which work well on medium resolution imagery of urban areas cannot be directly applied to high-resolution imagery [37]. This is due to the fact that the image content varies considerably for the same scene as the resolution increases. For example, the vertical sides of buildings, if visible, begin to form significant image areas; details of the building surroundings are visible and are usually not homogeneous; and fine structures on roofs, such as chimneys, become apparent and exhibit strong contrast with the roof itself. Standard edge detectors perform poorly in such situations, producing strong edges belonging to fine structures, as well as fragmented and irregular edges belonging to the roof boundaries.

Number of Images of the Scene and Types of Imagery

Monocular imagery affords a single image of the scene under consideration. Two or more images of the same scene from different angles greatly improves object observability, firstly, by allowing height data in the form of a digital surface model (DSM) to be generated through stereo matching, and secondly, by offering the possibility that an object which is occluded in one image may be less occluded in another image from a different angle. Source imagery may be of various types such as greyscale, colour, and multi-spectral.

Other Data Sources

Increasingly airborne laser scanners are used to provide accurate DSMs [24]. Digitised maps, cadastral maps and geospatial databases also provide valuable knowledge to aid the extraction process if they are available. Fusing the information provided by the above-mentioned data sources together with that offered by source imagery is an important and growing research area [38, 39].

Image Quality

Image quality and the suitability for building detection can vary considerably, depending on the conditions at the time of acquisition of the imagery. Factors affecting image radiometry (contrast) include the sensor view angle, the sun angle and resultant shadows, the seasons and atmospheric conditions [40].

Sensor noise and other artifacts also affect the image quality. These artifacts are introduced by the particular sensors used, the image acquisition process, and the follow-on processes (analogue film scanning, digital image compression and so on) that are performed prior to the imagery being made available for use.

In terms of object observability, height data is a key factor irrespective of whether it is derived from stereo matching, airborne laser or some other means. This is because such data allows for the direct verification of buildings (although, they need to be distinguished from other above-ground objects such as trees). If no height data is present verification of 3D objects must be inferred from cues such as shadow, which is often more challenging.

2.5 Models

Building detection system designers choose to model various aspects of the problem and these models vary in the extent of their detail, from rudimentary to sophisticated. First and foremost is the object model, that is, the model of the buildings that are being searched for. Additionally, there might be models of the scene and the sensor. These models are discussed below (adapted from [17]).

2.5.1 Object Model

Object models encode information about object appearance, shape, physical properties and so on. The model has to be flexible enough to allow for variation in the

presentation of the object due to viewpoint, illumination and differing members of the object class [25] but also rigid enough to demarcate a certain class of objects.

Ideally, object models should be described in the world domain, sometimes also referred to as the object space, rather than in the image space [41]. For example, object dimensions should be expressed in metres rather than pixels (an image space representation). Object space representations allow for multiple estimates of feature positions from different images, integration with cartographic information, and so on. Image space representations vary for a given object depending on the image scale and have no universal frame of reference for integrating with other data sources.

Many modern building extraction systems utilise strong *structural* models. A structural description of an object involves describing the object as an organisation of sub-objects or constituent elements [16]. Hierarchical object models are recursive in that each sub-object, once again, has its own structure, which can be described. This continues down to the level of primitives.

Important facets of object models for buildings are:

Geometry and/or Radiometry

Buildings tend to exhibit strong geometric regularities such as parallel sides and orthogonal corners. Detection systems which identify buildings based on these regularities are said to be using a geometric object model. Often such systems are edge-based and group edges according to expected regularities. Radiometry refers to the brightness, contrast and homogeneity of the visible building surfaces in the image. Radiometric-based models are formulated in terms of pixel intensities and often use the uniformity of image regions forming the roof surface as a criterion for identifying buildings. A number of approaches combine geometry and radiometry (see chapter 3).

2D or 3D

Object models can be defined in two or three dimensions. For example, 2D polygons may be used to delineate the boundary of the roof surface or ground plan, while 3D prisms or polyhedrons may be used to reconstruct buildings if height data is available or can be derived. It is also helpful to view object models as modelling two separate aspects: the building footprint and the building rooftop shape [42].

Kind of Representation

Different types of buildings models are used. *Parametric* models assume a

given building shape, the exact dimensions of which are governed by a number of free parameters. An example of a two-dimensional parametric model is a rectangle with the parameters of width and length. An example of a three three-dimensional parametric model is a building with a rectangular ground plan and a symmetric, sloped roof. The parameters here are the width, length and height of the box and the height of the roof ridge. A common strategy that is employed with parametric models is to apply the model to building parts rather than the entire structure. The building-part models are then grouped to model the complete building. In two-dimensions this would mean a building could be represented by, for example, a grouping of rectangles. Refer to chapter 3 for a more detailed discussion on two-dimensional strategies.

Models may also be *generic*, for example, polygons in 2D and prisms or even polyhedrons in 3D. Generic models allow for a greater variety of buildings to be modelled but at the same time they do little to discriminate between building and non-building objects. More specifically, parametric models are better able to separate buildings from non-buildings at the cost of an increased number of false negatives [43].

Both parametric and generic models have advantages as they make different trade-offs and it is suggested that they be integrated in some manner [43]. It is also worth noting that the system presented in [44] utilises both kinds of 3D models. Here, the different model types are not integrated, instead, an internal evaluation is performed to select the best-fitting model type for each hypothesis.

Level of Detail

The object model may range from being detailed, wherein, for example, building sub-parts or the reflectance properties of the roof material are modelled, to being very simple in the case where buildings are modelled as 2D rectangles.

2.5.2 Scene Model

The scene model, in the context of a building detection system, refers to any aspect of the scene that is modelled besides the buildings themselves. Scene modelling naturally includes the modelling of trees and roads but also, importantly, includes the modelling of shadow.

2.5.3 Sensor Model

The characteristics of the sensor itself may be modelled as well as aspects relating to the acquisition of the imagery, such as the sun vector at the time of acquisition. From a detailed 3D object model, it is possible to determine how the object will project into the image domain given knowledge of the sensor's viewing angle. This knowledge is crucial if the system allows for greater generality in the viewing angle, for example, oblique and nadir views as opposed to nadir views only.

2.6 Detection or Extraction Strategy

A substantial part of any vision system is the logic which is used to search and find the target objects, that is, the detection or extraction strategy. The strategy is applied to the source data and makes use of the relevant models in order to detect and possibly reconstruct the buildings in the scene. Important points pertaining to the strategy are given below.

2.6.1 Structural, Statistical and Hybrid Approaches

Detection of objects using a structural strategy is applied when the object models are structural. Grouping techniques are used to aggregate primitives or sub-objects having certain desired attributes and relationships between one another to form hypotheses. Grouping may take place in two and/or three dimensions. Grouping, as mentioned earlier, is an important mid-level vision process that often dominates building detection strategies for monocular and other imagery (Chapter 3).

Statistical approaches (as the term is used in [16]) use a statistical or decision theoretic approach to building detection and are typically employed with multi-spectral imagery. Individual pixels are classified according to their feature vectors. Pure statistical approaches lead to severe limitations as spatial and contextual information is completely ignored.

If the appropriate source data is available it is possible to integrate statistical and structural approaches. This results in a hybrid model which inherits the advantages of both types of techniques. Hybrid extraction strategies have been shown to offer superior results to either structural or statistical strategies on their own [35].

2.6.2 Control

Strategies may be *data-driven* or *model-driven* or a mixture of both. Model-driven strategies are typically used when the scene to be interpreted has a definite structure, such as a chest x-ray. In such cases, a great deal of knowledge is known *a priori* concerning the structure of the scene such as approximate size, shape, orientation and position of the target objects. The target objects, for example, the ribs, can then be decomposed into sub-objects, which are further divided into a configuration of primitives. The image can then be interpreted by trying to find appropriate primitive configurations. In other words, the detection strategy can be driven by the object model which expresses domain-specific knowledge in terms of the final goals of the system. The model is able to dictate where in the image to search, and what primitives or sub-objects to expect. This approach is also known as top-down processing or goal-driven processing.

In contrast to this is a data-driven or bottom-up analysis strategy. This involves starting at the lowest level of the representation hierarchy and identifying primitives. These primitives are then grouped into more meaningful entities. As one migrates up the representation hierarchy from primitives to complex objects, an increasing amount of domain-specific knowledge is brought to bear on the interpretation process.

A purely data-driven approach is subject to the following deficiencies [16]:

- Higher-level knowledge is excluded from the low- and mid- level processes, which results in a poor and sometimes arbitrary segmentation.
- Context is largely ignored in the early stages of analysis. Therefore, an extensive search process is required to locate regions of interest (regions in which object primitives are likely to be found) and ultimately the target objects.

Vision systems for aerial photography that are both model- and data-driven are able to focus attention efficiently and take advantage of knowledge from any level at any stage of the analysis [16, 25]. This hybrid approach is feasible, because although the scenes depicted by aerial images are highly unstructured, the buildings which they contain exhibit regularity of structure. In spite of the advantages of a hybrid approach, data-driven strategies are popular for building extraction systems [17] due to the unstructured nature of the scene content.

Building detection systems which do employ a hybrid approach often utilise a *hypothesise-and-verify* technique. A building hypothesis is formulated using a data-

driven approach. Assuming that the hypothesis is, in fact, correct, a specific search is conducted for additional evidence predicted by the object and/or scene model. The presence of such evidence is used to verify the hypothesis. Specific examples of this technique are discussed in the following chapter. Hypothesis formation may be decoupled from verification and rely on different types of evidence [45].

2.6.3 Fusion

Fusion is another important aspect of the strategy and refers to the combination of information from different sources. Fusion may occur at many levels, from fusing image data acquired by different sensors to fusing objects detected by different extraction strategies, to fusing image evidence for objects from different views of the same scene [46, 38, 47, 48]. Fusion in its various manifestations is shown to improve detection performance, although errors may also accumulate as shown in [46].

In cases where different extraction techniques are applied to the same image, the manner in which the extracted data is fused depends on the assumptions that are made about how the techniques work together. Liow & Pavlidis [49] suggest that a building recognition system should have specialised components, each of which provides a solution for a sub-domain of the overall task. In other words, different components should be responsible for finding different types of buildings. The results from each component can then be readily combined. Shufelt & McKeown [46], in contrast to this, introduce a cooperative-methods paradigm in which it is assumed that each building detection method employed will attempt to identify all the buildings in a scene but will be unable to do so due to inherent deficiencies. They present a simple, yet effective, method for accumulating the results of different building extraction techniques.

2.6.4 Internal Evaluation

Internal evaluation refers to the extent to which the strategy incorporates an evaluation of the outputs that are produced at different representation levels. This evaluation may take the form of probabilistic methods or other means of uncertainty modelling. The evaluation itself is used to resolve internal conflicts between different possible models, primitive groupings and so on. It may also be reflected externally to aid human photo-interpreters. For example, the confidence level of each building hypothesis may be shown to the user.

2.7 System Goals

The output of building detection systems, which are usually instances of the object model, vary widely depending on the envisaged application and the tasks that the system will be required to support [50]. Applications are diverse and include cartography, town planning, architecture, environmental analysis, and visualisation, among others.

For some systems, the goal is to automatically detect and *reconstruct* architecturally complex buildings in 3D. In other words, precise world domain descriptions of the buildings present in the input images are generated. Applications for such systems include the automatic production of highly detailed 3D city models for simulations [51]. These systems typically use stereo or multiple images, and in some cases airborne laser scanner data, and are becoming more commonplace [24]. A good overview is given in [42] and recent work in this area includes [52, 53].

Other systems, however, have less ambitious goals because the tasks that they support do not require accurate 3D building descriptions — for such systems the existence of a building is more important than its geometric description and an approximate building footprint or roof outline suffices [6, 54]. Many end-user applications which are related to the management of informal settlements have less stringent requirements on the system output, such as shack counting, change monitoring and GIS data updating, and have already been listed in the Introduction (Chapter 1).

One particular example is that of electrical reticulation planning, which applies both to informal and rural settlements. In planning an electrical network, spatial information regarding the location of consumer households is required for transformer and junction placement and cable routing [55, 56]. The spatial information that is required is simply a 2D point representing the centroid of a shack or building [15]. This can be derived from an approximate building footprint.

A very different task for which only building footprints are required, is the prediction of radio wave propagation from base station antennas for the analysis and design of wireless networks [57]. Two-dimensional ray tracing can be used to predict the reflections, diffractions and diffusion that occur as horizontal waves propagate. In this situation is desirable to simplify the building footprints by removing vertices in order to reduce the computationally-intensive ray-tracing calculations [57].

Finally, the use of simpler object models is being driven, in some cases, by readily

available, high-resolution (0.6–1 metre ground pixel) IKONOS and Quickbird satellite imagery. This is leading to new research efforts focused on imagery with a lower resolution than is attainable from aerial imagery. It appears that the extraction of detailed 3D building models from satellite imagery is less achievable due to the lower resolution and consequently, weak object resolving power [58]. Therefore, the outputs from such imagery tend to take the form of pixel highlighting or 2D polygonal/rectilinear footprints in the image space [35, 14, 29, 12] which are suitable for tasks such as map generation and revision.

2.8 Evaluation of System Performance

Judging a building detection system's performance is crucial for evaluating different strategies and algorithms. Quantitative performance evaluation metrics for building detection are commonly *measures of discrepancy* in which the differences between the system output and a manually generated ground truth are calculated [59]. These differences may be calculated at various levels in the representation hierarchy from pixels to object models. Performance metrics based on such calculations broadly mirror subjective relative visual estimates of quality, although there is debate as to how well they are correlated [59].

Performance metrics identify four different categories:

1. True Positives (TP) represent buildings identified by the detection system and labelled as such in the ground truth.
2. True Negatives (TN) represent non-buildings or background identified by the detection system and labelled as such in the ground truth. For systems with a binary classification "background" is not explicitly detected but deemed to be composed of everything other than the detected buildings.
3. False Positives (FP) represent buildings identified by the detection system but not labelled as such in the ground truth, that is, erroneous detections or errors of commission.
4. False Negatives (FN) are buildings that are not identified by the detection system but are labelled in the ground truth, that is, missed detections or errors of omission.

The following metrics are presented in [60, 35] and are widely used:

$$\begin{aligned} \text{Branching Factor} &= \frac{\text{FP}}{\text{TP}}, \\ \text{Miss Factor} &= \frac{\text{FN}}{\text{TP}}, \\ \text{Detection Percentage} &= 100 \cdot \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{Quality Percentage} &= 100 \cdot \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}. \end{aligned} \tag{2.1}$$

The detection percentage simply determines the percentage of buildings that have been detected by the system. The quality percentage is a stricter measure, factoring in the effect of false positives. A system can only obtain a quality measure of 100% if it neither misses buildings (FN = 0) nor falsely identifies any (FP = 0). The branching factor represents the degree to which an extraction technique incorrectly classifies background as building. The miss factor is a measure of how much building is missed by the system relative to that detected.

Shufelt & McKeown [46] determine TP, TN, FP and FN by comparing the output image that their system produces with the manual ground truth on a pixel-by-pixel basis. However, these metrics can be equally applied at a higher representation level to the building hypotheses produced by the system as compared to those identified in the ground truth [48, 61]. When calculating the metrics using building hypotheses rather than pixels, a decision needs to be made as to what constitutes true positives, negatives and so on. In some systems a building is considered detected, that is, a true positive, if any part of it is detected [48, 14, 62]; in others the contribution of partly detected buildings to the TP score is reduced [36]. Slight modifications to the above equations have been published [12, 48] but the resulting metrics are conceptually similar to those given here.

The shape accuracy of a system can be roughly gauged for the entire image by the quality percentage metric if it is calculated on an area/pixel basis. Alternatively, 2D building-by-building shape accuracy statistics can be generated by considering the extent of overlap between each detected building and its corresponding ground truth [63] and the similarity of moments between the extracted and reference buildings [62]. Positional accuracy of salient building features such as corners can also be used as a means of quantifying accuracy [12, 62].

Some of the above evaluation techniques have direct counterparts in 3D, but other more sophisticated 3D techniques can also be used. These are not discussed further

as they are beyond the scope of this thesis.

In spite of performance metrics, it is difficult to compare the absolute performance of systems due to one or more of the following factors:

- The source imagery to which different building detection systems have been applied, and for which results have been published, varies considerably. Source data directly affects the difficulty of the extraction task (Section 2.4) and hence the performance on specific types of images cannot be expected to hold generally. Note, there are some standard test datasets which are publicly available such as those from the Ascona workshops [64] but these do not, in any way, cover the wide range of building types and scenes for which detection systems have been designed. Furthermore, test imagery that is available often lacks the accompanying ground truth.
- The system goals can range from stringent demands on accuracy to part-detections being deemed acceptable, and the above metrics (Equations 2.1) may not fully reflect the system's performance in light of the applications it is required to support.
- The metrics used in evaluating system performance sometimes differ slightly.

A review of the performance of different building detection systems in given in [14]. The detection percentages and branching factors range widely for the systems reviewed (detection percentage: 41.5% to 97.6%; branching factor: 0.0% to 46.0%). The authors state that comparing the reported performances is subject to *caveats* similar to those listed above.

2.9 Conclusion

The computer vision field of building detection is an important one and has generated a significant amount of research. A number of high-level computer vision concepts, as well as more specific building detection concepts, have been described. These act as a useful lens for viewing and understanding different systems. At a detailed level, building detection systems vary significantly along many dimensions, including: the types of source data used, the scenes for which the system has been designed, the detection approach, and the object model and system goals.

The following chapter builds on the background provided in this chapter and gives a detailed review of existing systems which are relevant to the work at hand.

Chapter 3

Review of Existing Building Detection Systems

The work presented in this thesis is located at the intersection of building detection systems which are designed to operate on single images, systems which are dedicated to the detection of individual shacks in informal settlements, and systems which perform building detection within the context of a scale-space. In this chapter representative approaches from each of these areas are categorised and reviewed. Additionally, various techniques for boundary modification, which are key to systems using region primitives, are discussed. Modification techniques are shown to have two main goals — localisation and regularisation.

3.1 Introduction

In this chapter reviews are given of building detection systems that are relevant to this thesis. The system presented here sits at the junction of three categories of systems: systems which use single or monocular images as their source data, systems dedicated to the extraction of individual shacks from informal settlement images, and systems which make use of a scale-space. This is illustrated in Figure 3.1. The defining characteristic for each of these categories is different. For "Single Image Systems" it is the source data; for "Systems for Shack Detection" it is the system goals; and for "Systems Using a Multi-Scale Strategy" it is the extraction strategy. Each of these characteristics are a key consideration when designing a vision system.

The following sections are devoted to reviewing building detection systems residing

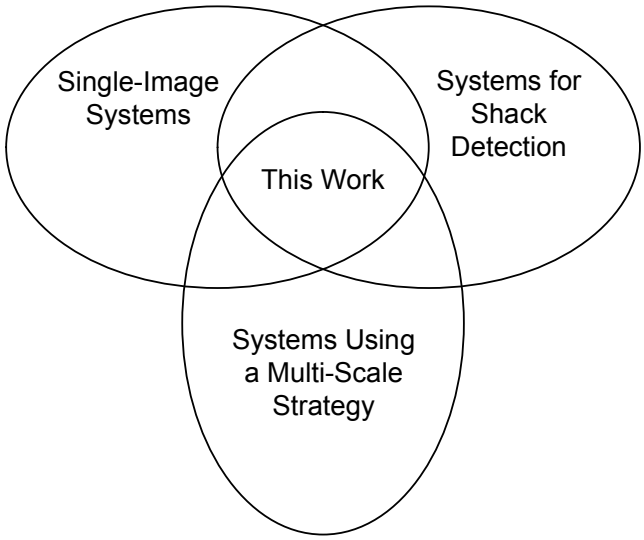


Figure 3.1: Categorisation of building detection systems relevant to this work.

in each of the above categories. An additional section is provided discussing issues around boundary localisation and regularisation as this will prove important in understanding the strategy that has been developed.

Although the focus here is on interpreting aerial imagery, systems which utilise high-resolution satellite imagery are reviewed where relevant. This is appropriate because such imagery can largely be treated as aerial optical imagery [24]. Additionally, high-level, knowledge-focused techniques are not reviewed as the concern in this thesis is primarily with the investigation and development of medium-level processes within the context of a scale-space.

3.2 Single-Image Systems

A "single-image system" is one that utilises a single or monocular intensity image as its data source. This section reviews systems that predominantly use greyscale nadir-view images although some systems dealing with colour images are included. Single image systems can be categorised according to whether the detection strategy primarily makes use of edge or region primitives, or a combination of the two (a hybrid strategy).

Regardless of the category, the use of shadow is a key component in many of these

systems. This is because shadow is an obvious, and relatively easily detectable, cue for 3D objects. Shadow finding allows further processing techniques to focus attention on specific regions of interest in the image. This reduces the search space, which implies a corresponding reduction in processing time. Shadows can also provide a large amount of information about man-made structures without ever having explicit models of the structures [45]. Additionally, if the sun inclination angle is known then the length of cast shadows can be used for height estimation. The use of shadow for building shape prediction, grouping, verification and height estimation is explored in [45].

3.2.1 Edge-based Strategies

This section describes systems employing edge-based recognition strategies. The typical procedure in these strategies is the application of an edge detector (for example, the Canny operator [65]), followed by straight line extraction [66, 67, 68], as it is assumed that building boundaries are composed of straight lines. Straight line extractors prune the large set of edges found by the edge detector to a more relevant subset which hopefully forms part of the boundaries of the objects being searched for. At this stage edges are grouped in certain configurations, ultimately to form complete building boundaries which undergo verification, usually involving the use of shadow in some manner. Specific systems are described below.

Huertas & Nevatia [32] present an edge-based technique for finding buildings. They require the monocular, greyscale source images to be of good quality and to depict scenes of medium density housing. It is assumed that man-made structures are rectangular or can be composed of rectangles. An edge image of the scene is constructed and linear features are extracted. Corners or pairs of line segments forming nearly orthogonal junctions are searched for and classified as either shadow or object corners. Object boundaries are traced by "chaining" object corners which share common segments and are consistent with the system model (a composition of rectangles). Finally, shadows are used to verify closed sets of chains or boxes, and in some cases, to complete the partial rectangle formed by open sets. This method is shown to work well if a high contrast exists between a house and its background, and if the houses have a simple roof structure and the density of buildings is low.

Mohan & Nevatia [69] use perceptual grouping for finding and describing complex buildings in aerial images. They use collated features to represent the structural interrelationships between primitive image features. The collated features are

constructed from the generic shapes of the objects being searched for. It is assumed that buildings can be modelled as a grouping of rectangles. Building shapes are successively broken up into rectangles, "U" shapes (open rectangles), parallel lines and eventually, straight lines. These form the collated features which are searched for. The system combines both bottom-up analysis and top-down analysis as there is an exchange of information between large collations and their constituent sub-collations. The system is edge-based with the starting point being an image containing line segments corresponding to significant edges in the original scene.

Lin & Nevatia [48] describe a system for detecting and reconstructing buildings from monocular images taken from either a nadir view or an oblique view. This is one of the more recent works in the area of single-image systems, building upon the work in [70, 69, 32] and the system itself is sophisticated and makes use of a sensor model. Their approach is summarised in some detail below.

Building roof shapes are restricted to parallelograms or their compositions based on the observation that roofs of rectilinear buildings project to parallelograms in 2D under weak perspective projection, which is seen as adequately modelling most aerial imaging situations. Lin and Nevatia use both geometric and projective constraints to construct building roof hypotheses from low-level features. These hypotheses are verified using local 3D cues.

Their system is edge-based as they first process the source image with the Canny edge detector and then use a linear feature extractor to extract significant straight edges. These features undergo additional processes aimed at further reducing the number of fragmented line segments. The result of these processes is a set of edges and junctions.

Parallelogram formation then takes place. The constraints for a particular image are derived from photogrammetric information about the imaging geometry, namely, the camera tilt angle and swing angle. Antiparallel edges (parallel edges of opposite contrast) are grouped together in pairs, subject to a variety of criteria based solely on edge geometry (length, position, relationship to other edges) and the projective constraints. These antiparallel edge pairs are called trigger edges. A search is then conducted for the missing sides of the parallelograms. All suitable side candidates are kept with the trigger edges being extended, as necessary, to connect with the side candidates and form parallelograms. This means that a number of parallelograms may be generated for one pair of trigger edges.

The set of hypothesised parallelograms that is generated is pruned in order to try

and eliminate false positives. This is done by firstly evaluating local criteria followed by global criteria. Local criteria are divided into positive and negative evidence. Examples of positive evidence include the percentage of edge coverage along the perimeter of the hypothesised parallelogram, and the number of matches between the potential shadow casting corners of the hypothesised parallelogram and the potential shadow corners extracted from the image. Examples of negative evidence include the displacement of edge support from the hypothesised parallelogram, edge support having T-junctions and/or L-junctions penetrating the parallelogram and the overlapping of parallel gaps on the perimeter of the parallelogram. All types of evidence are numerically scored, multiplied by a weighting factor and added to give an overall score. Hypotheses scoring less than a certain threshold are removed, leaving "selected hypotheses".

The hypotheses or parallelograms that remain are then evaluated according to global criteria in order to remove certain cases of overlap. At the global level, entire hypotheses are compared to one another. Duplicated hypotheses, evidentially contained hypotheses (where the supporting evidence for the contained hypothesis is completely subsumed by the supporting evidence for the containing hypothesis) and contained hypotheses are resolved.

Finally, verification of the selected hypotheses takes place and 3D shape is inferred. Verification takes place in the following manner. For each hypothesis a number of building heights are postulated. For each height wall and shadow evidence are accumulated, and the height at which the best combined evidence value is obtained, is selected. Knowledge of the viewing angle is used to project wall heights and knowledge of the sun angle is used to narrow the search for shadow evidence. The wall and shadow evidence are combined using a certainty factor method and the hypothesis with the height that gives the best confidence value is chosen. This confidence value is again combined with the roof evidence score (the score for the selected parallelogram), which was derived earlier, to give a total confidence value for each hypothesis. If this value is greater than a given threshold the hypothesis is considered verified. Occlusion analysis is performed now that all hypotheses are in 3D and it is known from the viewing angle which hypotheses could be expected to be occluded. This may result in new hypotheses being verified.

Lin and Nevatia note that they allow the user to choose the confidence level at which to display the results. There is a trade-off between the detection rate and the number of false positives. The authors use intuitive and heuristic evaluation functions for scoring items of evidence. A linear weighted sum is mostly used for

combining evidence with the weights being determined empirically. It is noted that the combination rules have been chosen on the basis of simplicity. In later work alternative methods for evidence combination and the learning of parameters, such as Bayes reasoning, is used [71].

Kim & Muller [36] propose a building detection strategy based on perceptual grouping through the construction of a line-relation graph. This approach is similar to earlier work described in [72]. Their strategy is divided into four stages: straight line extraction, line-relation graph construction, hypothesis generation and verification. Importantly, their entire strategy is based on geometric relationships between extracted lines — no radiometric properties for buildings are assumed. This allows their strategy to be less dependent on the intensity of the particular image being analysed.

Edges are extracted with the aid of an edge detector and approximated by straight lines. A graph is established with lines representing the nodes, and parallel and approximately perpendicular line relations forming directional graph edges. Building detection then takes place by searching for closed loops and 'U' shaped chains of lines in the graph. Building hypotheses sharing common graph edges are merged and hypotheses are verified through the use of shadow. Vertical shadow lines are first detected followed by the detection of ground-level shadow lines. If a building hypothesis contains a shadow line as one of its components or the line lies within its area it is regarded as a false hypothesis and eliminated. The authors also use vertical building lines to verify hypotheses in the case of oblique imagery. The system is shown to work fairly well with pixel-based detection percentages of 71% and up being reported on two test images, one being of University College London and the other being the ISPRS "Flat" test image (containing low density formal housing).

Krishnamachari & Chellappa [73] present an energy function based approach to detecting buildings in greyscale aerial images. Straight lines are extracted from an edge image as in [69]. A Markov Random Field (MRF) model is then built on the extracted lines. The model is chosen to support the properties of rectangular shapes. The energy function associated with the MRF is minimised allowing the lines to be grouped into rectangular structures. Finally, a deformable contour is used to complete partial rectangles. It is assumed that buildings are flat-topped and rectangular.

3.2.2 Region-based Strategies

These detection strategies are based around the extraction and classification of image regions. Regions may be extracted from the source image in different ways. These are described below.

One method of forming regions involves thresholding, a pixel-based measure, followed by connected component extraction. For example, fractal-based measures have been used in an attempt to discriminate between man-made objects and natural vegetation [74, 75]. These measures demonstrate some success in differentiating between deciduous vegetation and buildings but fall short when the natural scenery is composed of bare patches of land with a homogeneous appearance, as shown by the author in [76]. Thresholding is also often employed when height data is available. An altimetric threshold is used to identify all regions in the source data which are a certain height above ground level and these regions form the basis for building hypotheses, as in [77, 78, 79].

Another method of region formation is to use a segmentation algorithm to segment an entire image into regions. These algorithms group neighbouring pixels together if they meet some similarity measure. The initial partitioning of an image into regions typically results in over-segmentation. In order to rectify this regions may be merged according to various criteria including similar statistical properties, domain-specific knowledge, and knowledge about the segmentation process itself. Regions may need to be split when the variance of one or more of the properties of the region is too great. Split and merge algorithms refine segmentations by repeatedly splitting and merging regions according to built-in criteria until no further splits or merges take place. The regions then need to be classified in order to identify those corresponding to building roofs. The latter parts of this section review how particular region-based strategies tackle image segmentation and region classification.

Irrespective of the manner in which regions are formed, an important component of region-based strategies is the localisation and regularisation of region boundaries which are thought to correspond to buildings. This is necessary because region boundaries tend to be irregular and sections of the boundary may exist which map to intensity changes in the image but which do not correspond to the boundary of the building roof being delineated. This is common in cases where there is little contrast between sections of the roof and the surrounding background, and where there is uneven illumination of the roof surface or the surface varies in appearance. Boundary localisation and regularisation is discussed in detail in Section 3.5.

Note, this is not a major issue with edge-based strategies because straight (regularised) lines are extracted early on in the process, often by approximating edges with a straight line joining their endpoints. Additionally, these edges tend to be well localised because edge operators identify well defined edges; edges which are less discernable have magnitudes that fall below the operator's threshold and are filtered out, leading to edge fragmentation. So, for edge-based strategies the grouping together of edge fragments belonging to a particular building's boundary is more of a concern than the localisation and regularisation of a given closed boundary. The following paragraphs review building detection systems which are representative of region-based segmentation strategies on single images.

Paparoditis et al. [37] present two approaches for building detection and reconstruction from mid- and high-resolution aerial and satellite imagery (between 10 cm and 1 m per pixel). The images depict scenes of suburban areas having low density housing and large rectilinear buildings with flat roofs. In the one approach, an independent interpretation of one of a pair of stereo images is undertaken, followed by 3D computation and matching of building co-ordinates using projective geometry and both images. Buildings are assumed to be polyhedral.

The "monocular" interpretation of one of the images proceeds as follows. Contours are extracted from the image and closed, leading to an adjacency graph of regions. A polygonal approximation of the region boundaries is performed. These polygons are then checked to see if they form parallelograms. The ones that do not are split up into parallelograms resulting in new regions being created in the adjacency graph. Each region is characterised by a feature vector having four components: area, average grey level, quality rate of the region boundary approximation, and compactness. These feature vectors are labelled using a neural network as one of six classes, including a "building" class. Constraint propagation is used to ensure that region labels are consistent with the labels of surrounding regions. Their results have a significant number of false negatives due to a poor initial segmentation and classification.

Rodriguez et al. [80] have designed a vision system that identifies man-made objects based on their appearance in aerial images. In the monocular, greyscale source imagery rectangular buildings are seen to be composed of three main parts - two roof panels and the building's shadow. Each of these constituent elements has associated geometric, photometric and topological attributes. These building components form the "appearance" model which is defined within the image space. Appearance models are determined from a manual ground truth segmentation of a nominal image and these models can only be applied to images taken under similar conditions and

containing similar buildings. However, they avoid the explicit transition from object space to image space [17], and can be made more robust by relying mainly on geometric properties rather than radiometric properties [36, 78].

The actual recognition strategy takes place as follows. The source image is preprocessed and then segmented into connected components. The segmentation is based on intensity differences - two 4-adjacent pixels are assigned to the same component if the difference in their intensities is less than a certain threshold. From this original segmentation a "segmentation" hierarchy is constructed with connected components being progressively merged, based on border contrast, until the entire image corresponds to a single connected component. The system then searches through the hierarchy, using the appearance models to find instances of man-made objects.

Shape analysis is performed on all instances that are found. Roofs are assumed to have a rectangular appearance, therefore the component regions thought to comprise the rooftop are approximated by a 4-sided polygon. Regions which cannot be adequately approximated are rejected outright. For those that remain, a finer level of segmentation in the segmentation hierarchy is considered in order to more accurately delineate the object. All connected components at the finer level having a certain percentage, or more, of their area falling within the polygonal approximation are united to form the seeds for a new combinatorial search which attempts to find the largest combination of these components that coincide with the approximation. The appearance properties of the selected combination are again validated before accepting the combination as the hypothesis for the building in question. The results presented show that this approach is capable of detecting stand-alone, rectangular buildings with a simple architecture.

Liow & Pavlidis [49] present two different strategies for building detection which are tested on monocular, greyscale aerial imagery of medium to low density buildings with generally flat roofs. The one strategy is primarily region-based and is described below, while the other strongly integrates edges and regions and is described in Section 3.2.3. In both strategies it is assumed that building roofs appear as areas of nearly uniform intensity. Results are given demonstrating that these strategies are capable of identifying the majority of buildings in the test scenes.

In the strategy that has region growing as its primary process, buildings are not limited to having straight boundaries and may have curved borders. The first step is to apply a split and merge segmentation algorithm based on a uniformity criterion. The result of this step is an image which has been completely segmented into

regions. Each region is then checked to determine if it corresponds to a building's roof in the following manner. Every region is assumed to be a building and a check is performed along the sun vector direction to see if the hypothesis has a corresponding shadow. Hypotheses without shadows are eliminated. Then the shape of each hypothesis is checked for regularity. Direct shape analysis cannot be performed because of segmentation artifacts that occur, such as blobs. Shape analysis is indirectly performed by approximating each region discovered with a polygon. Perpendicular bisectors for each polygonal segment are created and the intersection points of these bisectors with other sides of the polygon are determined. If the length of any bisector (from one intersection with the polygon boundary to another) is less than a certain threshold then the hypothesis is discarded, as it does not form a regular shape.

The final stage of this strategy is to regularise and localise the boundaries of the remaining hypotheses. This is done in order to remove segmentation artifacts and to shift contours closer to the true positions of the edges in the image. Contours are shifted using a "snake-like" approach, initially formulated by [81]. If it is assumed that building boundaries are rectilinear then boundary segments are approximated by straight lines as described in Section 3.5.

3.2.3 Integrated Strategies

Integrated strategies, as defined in this thesis, make joint use of both edges and regions early on in the detection process. This definition therefore excludes region-based strategies which make use of edges in the final stages of the detection process to either localise or regularise the region boundary. In these cases, the strategy is seen as being predominantly region-based (Section 3.2.2) with edges being used to fine tune the results.

Liow & Pavlidis [49] also present an alternative strategy to that described in Section 3.2.2 which involves edge detection followed by region growing. In this approach, it is assumed that buildings have boundaries composed of straight lines. From a gradient image straight lines are extracted. Only shadowed lines (lines representing building edges that have adjacent shadow) are kept from this process as they have a high signal-to-noise ratio and are seen as reliable features. These lines are then grouped together in perpendicular and parallel or collinear configurations to delineate the shadowed boundaries of the buildings in the scene.

In order to complete the buildings' outlines, a region growing process is used. Building borders are estimated from the shadowed boundaries and all pixels within these borders, and within a certain intensity range, are marked as "building". The region growing operation experiences difficulties when there is low contrast between the building roof and the surrounding regions. In these situations, flooding occurs from the roof region to the surroundings. Morphological erosion and dilation with an asymmetric structuring element is used to fuse leaks on the non-shadowed sides of the region boundary. However, leaks remain if they are larger than the structuring element. These are identified as lying between two points on the boundary which are spatially close, yet have a large arc length between them. These leaks are removed by simply joining the two points with a straight line. Finally, building boundaries are localised and regularised as described in Section 3.5.

Henricsson [82] uses contours with associated colour region attributes ("attributed contours") for generating hypotheses for generic roof parts in order to reconstruct buildings in 3D. This review focuses on briefly describing these contours rather than the detection strategy within which they are used, as it is not relevant to the work at hand. The author makes the general assumption that an object surface is uniform in appearance along its boundary. This motivates the extraction of edges and lines which have associated narrow, flanking regions with photometric and chromatic attributes. These contours then undergo grouping based on the similarity of their attributes. However, the assumption of uniform appearance may be violated if the flanking region is disturbed by "noise", for example, the presence of a chimney or shadow in the flanking region of a rooftop edge. An estimation procedure is therefore used to remove outliers from the colour space when calculating region attributes. This results in noise, texture and other disturbances being removed from these regions and contours belonging to the same object having similar attributes.

3.3 Systems Using Imagery of Informal Settlements

The section reviews computer vision systems dedicated to the detection of individual shacks in informal settlement imagery[1]. The number of such systems is small in comparison to those created for extracting formal buildings from imagery of urban, suburban or industrial scenes. However, the research in this area appears to be on the increase, with much of the literature having been published in the last decade.

[1]Systems also exist which have the goal of mapping informal settlements at a regional level, rather than mapping individual shacks and buildings.

Owing to the limited number of publications in this area, the review here is broadened from focusing on systems with only a single aerial image as source data to considering systems utilising a variety of source data, including colour aerial images, satellite images and images with associated DSMs. Additionally, both automated and semi-automated systems are discussed. All systems equate shack detection with shack roof delineation in the nadir-view imagery that is used. These 2D outlines of shacks are useful for a variety of end-user applications, as explained earlier.

These terms, "semi-automated system" and "automated system", are used in different senses in the literature so they are defined here for the context of this thesis. A *semi-automated system* is one which requires user interaction for the *initial* identification of individual shacks. This typically involves manually clicking on part of a shack's roof. An *automated system*, on the other hand, detects shacks without user intervention. These terms, as used here, refer solely to the shack detection process. Pre- and post-processing steps may include user-controlled operations but it is the degree of automation of the shack detection and delineation process itself that is the defining issue. Examples of pre-processing steps, include the generation of a DSM, image pre-processing or other manual setup operations. An example of a post-processing step is the post-editing of delineation results to correct inaccuracies.

3.3.1 Semi-Automated Systems

Baltsavias & Mason [79] document one of the first attempts at semi-automated and automated shack detection, with the goal being to extract 4-sided shacks given that this is the most common shack form in South African informal settlements. Colour imagery of the Marconi Beam informal settlement is used. This aerial imagery is part of a dataset that was gathered for the University of Cape Town's UrbanModeler project in 1996, using a KODAK DCS460c digital camera [83]. The ground pixel size for this dataset is 0.18 m.

Baltsavias & Mason's semi-automated shack detection strategy involves extracting attributed contours [82] (as described in Section 3.2.3). They note that these contours, which have associated colour attributes derived from flanking regions, appear to be less useful for generating complete roof hypotheses than in urban scenes. This is partly due to the fact that shack roofs exhibit strong internal edge contours at points where dissimilar roof materials adjoin. Additionally, the contours do not accurately delineate shack boundaries when there is low roof-ground contrast. However, contours demarcating roof-shadow edges are accurately extracted.

These roof-shadow contours (a subset of all contours) are identified by selecting contours with a high contrast between their flanking regions. Their results show that most free standing shacks are partially delineated on the two sides forming the roof-shadow boundary (given that the sun vector is not parallel with either side). A user then interacts with the image by identifying the shadow corners of shacks with a mouse click. If two relatively perpendicular contours are in the vicinity of the mouse click they are intersected to form the shadow corner of the shack hypothesis. If they already intersect, the vertex of their intersection forms the shadow corner. The opposite end points of the two contours form two more corner points, and the corner diagonally opposite the shadow corner is computed so that a parallelogram is formed by all four points.

Quantitative results are not given for the Marconi Beam image but qualitative results show that many shacks are only partially delineated due to the fact that the shadow contours are fragmented and those associated with the shadow corner do not extend along the entire length of the roof-shadow sides. The authors also note that some shadow corners have more than two contours in their vicinity and it is ambiguous as to which two comprise the hypothesis. Consequently, this approach may fail for closely clustered shacks.

Li [13] extends the semi-automated approach presented in [79]. An interactive shack delineation scheme, termed SESDA, is developed which involves *initial* user interactions for identifying shacks and *corrective* interactions for manually editing the hypotheses extracted from the initial interactions.

The interactive scheme relies on Henricsson's attributed contours. It is shown that the number of attributed contours derived from the source image is large with some being fragmented. Many are spurious as they delineate different shack roof materials (rather than just the roof border) and non-shack objects. However, by using other cues such the DSM blob boundary contours, or the contours from extracted shadow regions, many false contours present in the source image are eliminated whilst key contours for shack delineation are retained. In other words, the remaining contours are far more reliable indicators of shack boundaries. Additionally, the total number of contours significantly decreases, reducing the computational complexity of the problem.

SESDA allows a user to add a shack to the site model by clicking on one or more corners of a shack roof. After each click a parallelogram is formed in a similar fashion to that described in [79] using either shadow edge contours or shadow and blob edge

contours. The results clearly demonstrate that using both cues is more successful than solely using shadow edge contours (see Table 6.6). Blob edge contours appear to be particularly helpful in cases where shadow edges are fragmented, or only a single shadow edge side exists. Users are also offered the opportunity to apply corrective interactions wherein shack sides may be manually increased or decreased, rotated, translated and so on, within the constraints of the shack model (a parallelogram).

Mayunga et al. [12] present a semi-automated method for building identification in informal settlements, using QuickBird satellite imagery of Dar Es Salaam, in Tanzania. This work is based on their earlier work described in [84, 85]. The study area is located in the eastern part of the city which is characterised by informal development.

Their approach makes use of *snakes*. A snake is an active contour model whose vertices are iteratively moved in order to minimise its energy function [81]. The energy function is composed of different energy terms, each of which influence the snake's movement, based on either internal forces, image forces or external constraints. The internal energy of the snake depends on its intrinsic properties and relates to the amount of stretching and bending that is allowed. Snakes are typically encouraged to move towards strong image edges as it is assumed that these correspond to object boundaries. In the context of building boundary regularisation, external geometric constraints are used to encourage right-angled corners, collinearity and parallel sides [61, 86, 87].

In [12], images are pre-processed, using anisotropic diffusion in order to homogenise regions and enhance building edges. Snakes are initialised on the pre-processed image by the user clicking on the approximate centre of each building in the scene. A radial casting algorithm is used to project radial lines from this centre point at defined angular intervals. Snake nodes are initialised on these lines and the snake deforms according to the energy function. If the snake's final position poorly delineates the building then the user is given the option of re-choosing the centre point. The internal snake energy is based on continuity and curvature, while the image energy term is constructed so that the snake is attracted to edges. Interestingly, in this study, the authors forego the external constraint term in order to detect more complex building types. A consequence of this is that corner points are rounded in the results that are shown. A variety of metrics are used to evaluate performance and some of these are reported on in more detail in Section 6.8.

3.3.2 Automated Systems

Baltsavias & Mason's automated extraction strategy relies on the existence of a detailed DTM of the area being mapped. This enables orthoimages to be derived and used and a normalised DSM to be generated. Stereoscopic image matching is used to generated the DSM and this is normalised by subtracting the DTM. The 2.5D elevation blobs derived from this process are thresholded at a height of 1.5 m. This height is chosen to discriminate between shack roofs and other raised surface objects such as cars.

Multi-spectral classification techniques (described more fully in [13]) are used to classify pixels into five different object classes: bright roofs, medium roofs, dark roofs, shadow and ground. It is shown that the roof classes are not sufficiently distinct from the others, revealing the difficulty of shack-ground separation. This is attributed to the fact that the ground, which is mostly bare, is spectrally similar to some of the shack roof materials. However, the shadow class appears to be mostly well separated from the others, and this is used to identify shadow in the image.

The automated detection strategy is relatively simple. The blob regions extracted after thresholding the normalised digital surface model (nDSM), excluding the regions masked by shadow, undergo morphological opening to provide a "coarse delineation" of the shacks in the scene. Altogether 67% of the total shack roof area is identified (true-positive area) with each individual shack being covered to some extent by the extracted regions. However, the shack boundaries are not very accurately delineated and closely clustered shacks are extracted as aggregations.

Rüther et al. [61] present an automatic approach for extracting approximate shack outlines from informal settlement imagery. The inputs required for their system are orthoimages and a DSM which has been generated from image matching techniques. As in [79], elevation blobs in the nDSM are derived and altimetrically thresholded. The centre points of these blobs are projected onto the orthoimage and form the seed points of a region growing process which is constrained by edges. The boundary of each region that is formed represents the starting position of a snake on the image. The snake energy function which is used to optimally delineate shacks encourages the snake to be attracted to image edges, and to exhibit sharp turning points in the form of approximate right angles.

This system is tested on an image of the Marconi Beam settlement in South Africa as well as on images of Manzese, a suburb of Dar es Salaam, Tanzania. Viewing

the outlines of the shacks which have been detected, superimposed on the source image, it is fair to say that the results are mediocre. A fair percentage of shacks have been identified but their delineation is poor with many sharp turning points on the extracted boundaries being mislocated. The authors conclude that the inaccuracies in the final boundaries are partly due to the initial positions of snakes being poorly localised (which is evident from intermediate results). Other reasons include the fact that the adoption of a single mathematical model (as expressed in the snake formulation) for all buildings in the study area may not be sufficient, and that buildings are closely clustered. Quantitative results for this system are presented in Section 6.8.

Li et al. [5] focus on the development of a colour edge detector for detecting shack roof outlines. They define a fuzzy measure for gauging the similarity between two colours in the RGB colour space. This leads to a definition of colour morphological operators for dilation, erosion, closing and opening. Edge detection is performed by applying dual morphological operations to the source image, and calculating the colour similarity between the two resultant images, which is represented as a greyscale image. In the final stage, edges are extracted by thresholding the greyscale image.

This edge detector is applied to the Marconi Beam dataset. It is shown that extracted shack roof edges are affected by noise due to objects of similar colours in their vicinity. Interior roof edges are also present due to lack of colour uniformity on some roofs. Directed binary dilation is used to alleviate these problems and resulting edge image is thinned. The extracted edges mostly delineate shack borders although shadow edges and some non-shack edges are also extracted and there are edge gaps. No quantitative results are given. This works focuses on edge detection so there is no attempt at grouping edges to form closed polygonal hypotheses.

The commercial image analysis software package, *Definiens eCognition*, has been applied to satellite imagery of informal settlements in both South Africa and Brazil [9, 8]. This software allows for multi-resolution segmentation of a source image based on a homogeneity criterion [88]. This criterion is based both on the colour and the shape of image objects. Image objects (regions) are merged iteratively in pairs as long as the homogeneity of the resultant merged object does not exceed the threshold.

A “scale” parameter determines the threshold and, consequently, the size of the image regions that are extracted. By varying the scale parameter it is possible to

create a number of scale levels having fine to coarse segmentations[2]. The image objects at each level "know" about their neighbours as well as their superobjects (at a higher level) and subobjects (at a lower level).

eCognition offers various features that can be used in the classification of image objects. These range from texture measures for an individual region to features based on the topology of the image object hierarchy. Additionally, a number of different classification techniques can be used, including fuzzy inference. Membership functions can be defined to translate feature values into degrees of class membership and these can be combined using standard fuzzy logical operators.

Importantly, Hofmann's work shows that the resolution of IKONOS and Quickbird satellite imagery is insufficient for the detection of individual shacks although informal settlement areas can be identified.

Finally, it is worth noting that although all of the systems referred to in this section and Section 3.3.1 have been tested on scenes of informal settlements, the architecture of the buildings in such settlements varies. This is clearly illustrated in [61, 12] where the selected informal settlements in Tanzania are shown to differ quite substantially from those in South Africa. In the Tanzanian images, the informal settlement buildings tend to be constructed of uniform roof materials and have a more complex roof geometry (usually with roof ridge). In other words, they appear similar to buildings which have been formally planned. This is in contrast to a typical South African informal settlement where shack roofs are generally composed of disparate materials and are flat.

3.4 Systems Using a Multi-Scale Strategy

The term "multi-scale" is broadly used and often applied to computer vision strategies that are quite different in nature. A multi-scale strategy is often taken to refer to a strategy which involves the generation and utilisation of a family of images produced by smoothing the original image. This family of images or image stack is known as a *scale-space*. A scale-space has the important property that as the original image is increasingly smoothed, fine scale detail is eliminated producing images in which the information has been simplified [89] even though each image in the scale-space stack has the same pixel dimensions as the original (see figure 3.2).

[2]This is a "scale-space like" approach but the scales are generated through region merging rather than the application of a diffusion equation or morphological operator, as described in Section 3.4.1.

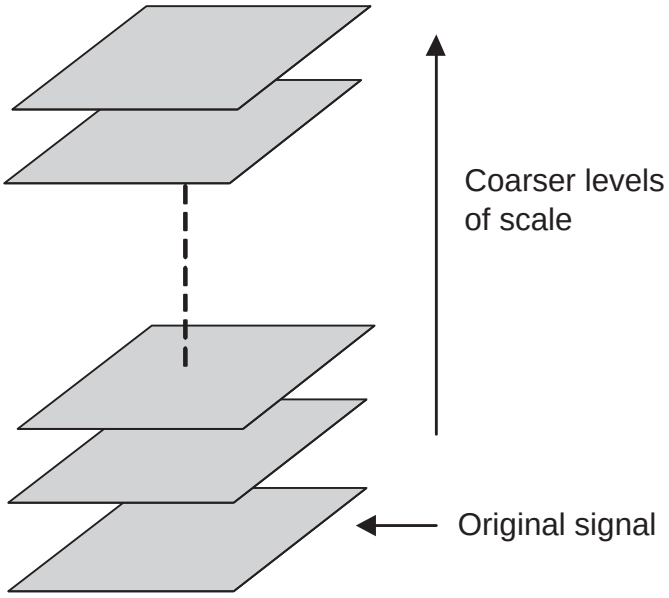


Figure 3.2: A scale-space stack (adapted from [90]).

Multi-scale techniques are different but related to *multi-resolution* and *wavelet* techniques [90]. In multi-resolution techniques the source image is also used to produce a family of images but these images are typically smaller than the source image. Multi-resolution techniques include the use of quad trees and the construction of image pyramids. Wavelets allow one to create an orthogonal multi-resolution representation of an image [91]. Multi-resolution and wavelet techniques, like multi-scale techniques, result in a reduction of image information. However, there are significant differences between these approaches and the generation of scale-spaces, both in the underlying theory and its practical implementation. Some of these differences are described in [92].

The term multi-scale is also used to describe strategies in which the scale of the feature detector varies rather than the image to which it is being applied. For example, in [93] a set of Gabor filters with different Gaussian widths are used to discriminate texture at several scales in the source image. In this case, the actual size Gabor convolution mask is varying. In the region-based systems, described previously, which construct a segmentation hierarchy through progressive merges [8, 9, 80], the homogeneity threshold of the segmentation algorithm varies resulting in a set of differing segmentations, all of which are derived from the same source image. These approaches are scale-space like in nature.

Finally, a vision system may use a classification hierarchy with classes existing at multiple scales or levels. In these cases abstraction occurs at a symbolic level rather than at a signal (image) level. For instance, in [94] the authors model the real-world

scene as a hierarchy of abstractions which serves as a control strategy for searching for objects. The image is initially divided up into superclasses which provide different contexts for different parts of the scene. Context-dependent searching then takes place for the objects predicted to exist at the next level of the hierarchy. An example from [94] is that a *Settlement* class is composed of different subclasses, including *BuildingArea*. *BuildingArea*, itself consists of *Building* objects and *Cast Shadow* objects. This type of approach is also prevalent in work using the eCognition software package.

As can be seen "multi-scale" is used in many different senses in the literature. In this thesis, a *multi-scale strategy* will be used to refer to approaches in which multiple simplified versions of the original image are produced to form a scale-space, as illustrated above in Figure 3.2, and the focus of this review is limited to these types of approaches. Different types of scale-spaces, linking across scales, and the identification of features and objects within the scale-space are briefly discussed in the next few sections. This is followed by a review of building detection systems which operate within the context of such scale-spaces.

3.4.1 Types of Scale-Spaces

Scale-spaces can be divided into two main groups: linear scale-spaces and non-linear scale-spaces [95]. A linear scale-space, constructed by convolving an image with a Gaussian kernel, is seen to be unique in that structures at coarser scales are simplifications of the corresponding structures at finer scales [90, p. 4]. The construction of the scale-space itself does not introduce artificial structure. Furthermore, a linear scale-space represents a visually neutral (uncommitted) front-end, in that the Gaussian kernel is linear and spatially shift-invariant and therefore not tuned to a specific detection task [96].

One of the drawbacks of using a linear scale-space is that all structure in the image is blurred over scale. This results in features like edges becoming delocalised over scale and it makes them harder to identify [97]. If it is known *a priori* that edges (or other features) are useful and need to be preserved over scale then it is possible to turn to non-linear scale-spaces.

A variety of different non-linear scale-spaces have been proposed, many of which are based on non-linear diffusion models [97] while others are based on geometric flows [98] or morphological operations [99]. The choice of scale-space depends on the

application at hand. The non-linear, edge-preserving, anisotropic scale-space used in this study is described in more detail in Section 5.2.1.

3.4.2 Linking Across Scales

Koenderink [100] refers to the "deep structure" of images — the idea that it is important to understand an image at all scale levels simultaneously and not as a series of unrelated smoothed images. In order to achieve this it is necessary to relate the information contained in images at different scales.

For image segmentation tasks, there are two, fairly common, types of approach for determining deep structure. The first type involves the use of critical points over scale [101]. A critical point is any location in a image where the gradient is zero, such as, minima, maxima, and saddles. The second type of approach involves linking pixels [102] or regions [103] from one scale level to the next, based on intensity similarity, proximity, overlap and other measures. Linking processes in general produce hierarchical, tree-like structures with many children objects at a finer scale being linked to a single parent at a coarser scale. Overall, the number of objects at a given level decreases as the scale becomes more coarse.

3.4.3 Object Classification and Detection

In order to perform object detection within the context of a scale-space, it is necessary to extract relevant features from images in the image stack and to apply classification techniques to these. This is similar to how one would detect objects from a single image — the main differences being that features may involve input from multiple scales [104] or be extracted from different scales based on some criteria [105]. Additionally, features may be derived from the relationships expressed in the image object hierarchy. If objects are classified at a particular scale independently of other scales, then thought needs to be given as to how to resolve conflicting classifications of objects that are linked together over scale. Control strategies may be devised that allow detection modules to operate on an image at a coarser scale in order to focus attention on certain areas and then to drop to a finer scale where more detail is needed [106].

3.4.4 Building Detection Within the Context of a Scale-Space

Previous sections in this chapter have reviewed building detection/extraction strategies which operate on a single optical image, possibly in conjunction with height data. This section reviews the relatively small number of building detection strategies which specifically utilise scale-spaces as described in Section 3.4.1.

Stassopoulou et al. [107] present a system for automatically detecting building boundaries in digital orthophotos. The results that are given are based on greyscale images of low to medium density housing. A Gaussian scale-space is used with σ's of 1, 2 and 4 pixels. The authors note that smoothing the image with a Gaussian of $\sigma > 4$ pixels results in edges which are too poorly localised to be of use [108].

The image is segmented at each scale, and edge and region information are combined as follows:

1. Edges are extracted from an input image at a given scale using the Canny operator.
2. Edges are linked by closing small gaps, using a parallel edge growing procedure, to form regions with closed boundaries.
3. Region statistics (mean intensity and variance) for each region at the given scale are determined. Each region is segmented into child regions at the next (finer) scale by recursively calling this procedure.
4. A one-way Analysis of Variance (ANOVA) test is performed on each set of child regions belonging to a given parent [107]. If the ANOVA test indicates that the child regions are sampled from the same normal population as the parent region then the parent region is retained and not split. If the ANOVA test indicates that the child regions belong to different populations then the child regions replace the parent region. Additionally, if the parent region is less than a specified size threshold no segmentation is performed. This avoids the production of overly small regions which are not meaningful for the application.

The end result of the above process is an adaptive tree decomposition consisting of regions which have been identified at more than one scale. The segmentation is performed using edge information but the decision to further segment a region is based on region statistics.

The segmented regions which have been produced by the adaptive multi-scale process are then classified as buildings or not-buildings using a Bayesian network. A polygonal

approximation of each region (see Section 3.5) represents the building boundary in nodes of the network. One node deals with the *rectangularity* of the approximation. Rectangularity is defined as the ratio of 90° corners to the total number of corners and varies from 0 to 1. A value of one would indicate a perfectly rectilinear shape. Another node in the network represents *polygon fit*. Polygon fit is a measure of how well the area of the polygonal approximation area, A, fits over the segmented region area, B. It is defined as the ratio $A \cap B / A \cup B$ and ranges from 0 to 1, with higher values indicating a more accurate approximation.

Multi-scale segmentation is shown to reasonably segment buildings from the surrounding background and work even in some cases where the contrast between the building and the background is low.

Morphological scale-spaces by reconstruction have also been applied to the building detection problem in urban scenes [35, 109]. These scale-spaces do not suffer from boundary displacement problems as the shape of the structuring element is not imposed on image structures. Disc-shaped structuring elements of varying sizes are used to produce different scales. The profile of how each pixel's intensity changes over scale can be determined. The discrete derivative of this profile is known as the differential morphological profile (DMP).

At each scale the DMP for each pixel is thresholded and those pixels whose DMPs exceed the threshold undergo connected component labelling. These components form building hypotheses that are further verified using shape criteria. The size of the components extracted roughly corresponds to the size of the structuring element used to produce the particular scale. This leads to a problem in that components extracted at fine scales identify both substructure and small buildings and it is difficult to discriminate between the two. In order to resolve this, scales are only generated using structuring elements that are larger in size than the expected scale of the substructure, and small buildings are identified using an alternative procedure based on spectral information and region growing.

Shadows can also be determined from the DMP as the profile not only provides information about structure size but also about the brightness or darkness of a region relative to its surroundings. Shadow is used to support a third method of building extraction.

Finally, it is worth noting that anisotropic diffusion has been used by some researchers in pre-processing images to facilitate building and shack detection [93, 12]. Diffusion is not used to create a scale-space but to produce a single enhanced image in which

noise has been suppressed.

3.5 Region Boundary Localisation and Regularisation

The approaches to building detection and extraction from aerial and satellite imagery are diverse as shown in earlier sections; however, common to many approaches is the fact that building hypotheses are produced in the form of closed polygons. These polygons delineate the building's roof if the detection is constrained to two-dimensions but may also represent a building's ground-plan if the final hypothesis is three-dimensional [78].

Polygons can arise in different ways depending upon the source data used and the detection strategy employed. The focus of the following discussion is on closed polygons arising from the vectorisation of region boundaries, as opposed to polygons formed from the grouping of line segments (as is the case with edge-based strategies). This is especially important in this work because hypotheses are initially extracted as regions, as explained in Chapter 5.

There are two key issues related to the modification of polygonal hypotheses — *localisation* and *regularisation*. Localisation uses the source data to improve the accuracy with which the polygon delineates the building's boundary. Regularisation enforces geometric constraints on the polygon in accordance with the system's object model, which include parallelism, rectangularity and so on.

A number of different localisation and regularisation techniques have been documented in the literature and a brief review of representative techniques is given below. A couple of these techniques form part of systems which have previously been reviewed and the reader is referred to the appropriate sections for a more complete description. For the rest, the necessary background is given here.

Four different applications of the use of snakes are reviewed because each of these is unique and directly relevant to this work in one way or another. In some systems the starting boundary is derived from DSM height data rather than image segmentation. Even though height data is used, the boundaries to be localised and regularised are created by a raster to vector conversion of a chain of connected pixels and therefore they exhibit similar characteristics to those derived from image segmentation. Additionally, the goals of improved localisation and boundary regularisation are similar, irrespective of the manner in which the starting boundaries are derived.

The manner in which snakes are formulated is briefly described in Section 3.3.1. Rüther et al. [61] and Mayunga et al. [12] have applied different snake formulations to images of informal settlements (see Sections 3.3.2 and 3.3.1). Rüther et al. [61] use snakes which are attracted to strong image gradients, and which are encouraged to form right angles — combining the tasks of localisation and regularisation. Their results are noticeably poorer than others which, as mentioned earlier, is attributable to the difficult task of shack roof detection (as opposed to the roofs of formal housing) and the low accuracy of the starting boundaries which are used. Mayunga et al. [12] do not include geometric shape constraints in the formulation of their snake model with the consequence that their snakes are used solely to localise boundaries and the final boundaries exhibit more rounding and less rectilinearity than those produced by other techniques.

Žunić & Rosin approach the problem in an interesting manner in that they do not use image forces for influencing snake movement [86]. Boundary modification is based strictly on rectilinearity (Section 5.5.1) and the distance from the original contour to the final one. This approach is viable for contours that are fairly well localised to begin with. Mayer's [87] use of snakes is arguably the most sophisticated, involving two optimisation loops which separately enforce geometric constraints and attract the boundary to image edges. Other heuristic optimisations, including the creation of new vertices along the boundary at points of low edge support, are used.

In [110, 78] a vertex shifting technique is used based on minimum description length (MDL). An attempt is made to fit a set of given boundary points to a prismatic model, whilst minimising the complexity of the model. Models with more free parameters are deemed to be more complex. Four neighbouring vertices of the starting boundary are considered at a time, with the two outer vertices being fixed and the two inner vertices being shifted or replaced by a single vertex subject to minimising the local description length criterion. Ten alternative configurations, based on expectations about building shape, are considered for each set of neighbouring vertices. A final processing step takes place after MDL-optimisation in which all polygon segments are grouped by building hypothesis and vertices are adjusted in accordance with more global geometric constraints such as parallelism. This technique stands out from the rest in its ability to explicitly deal with nested boundaries.

The work of Liow & Pavlidis has been reviewed earlier. Here a brief description is given of how they regularise the region boundaries produced from their integrated approach — edge detection followed by region growing (see Section 3.2.3). Wherever possible, sections of the region boundary on the roof-shadow side are replaced with

straight lines which have been extracted in the earlier edge-detection phase. This results in a boundary composed of both straight line segments as well as segments of 8-connected pixels (from region growing). Finally, the remaining 8-connected segments of the boundary are approximated by straight lines having the same orientations as the replaced sections, or where this approximation is not possible, the two endpoints of the unfitted segment are simply joined by a straight line.

Caelli, Stassopoulou and Ramirez [111, 107] extract boundaries from segmented regions (see Section 3.4.4) and calculate the curvature. Peaks in the absolute value of curvature are determined and if the peaks are greater than a certain threshold then the corresponding boundary points are labelled as corners. The boundaries are smoothed prior to the curvature calculation by applying a Gaussian filter. The peak threshold and width of the Gaussian filter are learned through supervised training. Corners are joined using straight lines resulting in a polygonal approximation of the contour.

The system is trained using buildings of a certain size and therefore the width of the Gaussian filter is optimal for that size. Larger scale boundaries require more smoothing to emphasise corners while smaller scale boundaries require less smoothing (over-smoothing a boundary eliminates significant corner points). In their current system the Gaussian width is fixed and therefore not optimal for buildings which are significantly different in size from the training set. Evidence for this is shown by the system's poor polygonal approximations of buildings which are roughly half the size or less than those used in the training set. The authors point out that the Gaussian width parameter could possibly be learned as well.

Gerke et al. [112] present a "rectangular-decomposition" approach to boundary regularisation. This approach is different from the others reviewed in that the segmented region (derived from a DSM) forms the input to the regularisation process as opposed to the boundary of the region. Invariant geometric moments are used to determine the parameters (width, length, position and approximate orientation) of an initial rectangular estimate to a segmented building region. This rectangle is rotated to maximally cover the segmented region [94]. A process of iterative refinement is undertaken as additional rectangles are added to or subtracted from the initial rectangle to minimise the differences in area between the polygonal approximation and the segmented region. This process terminates when the next rectangle to be added or subtracted is less than a minimum area. The initial rectangle is modified solely through the addition and subtraction of rectangles with matching orientations so the approximation is perfectly rectilinear. Orientation errors may arise if the

segmented region itself is slightly skewed with respect to the ground truth, as it is this region that forms the reference for rotating the initial rectangle. A mis-orientated initial rectangle results in a poorly localised final boundary.

Tables 3.1 and 3.2 summarise the different boundary modification techniques which have been described. The columns of the tables from left to right are as follows:

Technique

The type of modification technique is given along with a reference describing its application.

Scene

This column refers to the type of scenery to which the technique has been applied. In some cases images of industrial areas have been used. These scenes are similar to that of suburbia in that the buildings are well separated, however, the architecture is often simpler with many buildings having flat roofs.

Boundary Origin

In cases where height data is used, the boundary may originate from altimetrically thresholding a nDSM and directly represent a height contour, or the boundary may result from a segmented region in a height map image. In detection strategies which do not use height data, the starting boundary is formed by the pixels making up the boundary of a segmented (optical image) region.

Digitisation Noise Removal

The starting boundaries exhibit digitisation noise as they are generated from a raster representation in which pixels are either 4- or 8-connected. This quantisation of connection angles results in artifacts such as straight lines appearing as staircases. Accordingly, most techniques involve an initial noise removal phase (in some cases, noise removal is repeated after the boundary has shifted position). Noise removal usually takes the form of approximating the original boundary with a subset of its vertices. The method used in each of the techniques is referenced where possible. Noise removal and boundary simplification is discussed in more detail in Section 5.3.1. It is worth noting that erosion and dilation, which are used in some applications, do not remove digitisation noise as these operations are applied in the image domain and the structuring element is itself subject to such noise.

Vertices

This column relates the vertices of the final boundary to those in the starting boundary. There are three, non-exclusive, options:

1. Subset – vertices present in the final boundary are a subset of those belonging to the initial boundary. Note that the digitisation noise removal phase is not considered here. The technique is only marked as "Subset" if the localisation or regularisation phase involves vertex removal.
2. Shifted – vertices in the initial boundary are shifted to new positions in the final boundary. This typically occurs in an iterative manner when snakes are used and each snake node is moved in turn in order to minimise the snake's energy.
3. New – vertices are present in the final boundary which have no direct connection with vertices in the starting boundary.

Accuracy – Start and Final

This column gives a qualitative assessment of how accurately the starting and final boundaries delineate buildings or shacks in the source image.

Inclusion of Image Data and Localisation/Regularisation

Boundary modification techniques may manipulate the starting boundary solely on the basis of an assumed model. In these situations no image data (or DSM data) is involved in the process, and it is seen as one of regularisation. However, some techniques explicitly include data additional to starting boundary in order to guide its modification and improve the accuracy with which it delineates the image object. In these techniques, the modification process may only involve localisation or it may involve both localisation and regularisation, if geometric constraints are present in the process.

The techniques in Tables 3.1 and 3.2 involve different numbers of phases. Many techniques have two different phases: digitisation noise removal followed by localisation/regularisation. The snake approach in [87] can be seen as having three phases: digitisation noise removal, followed by two distinct snake phases, model-driven regularisation and model-and-data-driven localisation/regularisation.

The approaches in [87] and [110] appear to be significantly more complex than the rest. The complexity in [87] is due both to its multiple snake phases and the additional heuristic optimisations that are involved. The approach in [110] also consists of multiple steps and exhibits algorithmic complexity in the use of MDL as a principle for both local and global shape analysis.

Technique	Scene	Boundary Origin	Digitisation Noise Removal
Snakes, Rüther et al. [61]	Informal Settlement	Image region grown from seed point	Weidner-Förstner [78]
Snakes, Mayunga et al. [12]	Informal Settlement	Click point + radial-casting algorithm	Not applicable
Snakes, Žunić & Rosin [86]	Urban	Derived from DSM	Ramer [113]
Snakes, Mayer [87]	Industrial Area	Derived from DSM	Douglas-Peucker [114]
MDL, Brunn et al. [110]	Urban	Derived from DSM	Weidner-Förstner [78]
Boundary Section Replacement, Liow & Pavlidis [49]	Industrial Area, Airport	Image segmentation	—
Corner Detection, Caelli, Stassopoulou et al. [111, 107]	Suburban/Urban	Image segmentation	Gaussian smoothing of curvature
Rectangular Decomposition, Gerke et al. [112]	Suburban	Derived from DSM	Not applicable

Table 3.1: Summary of boundary modification techniques I.

Technique	**Vertices**	**Accuracy – Start**	**Accuracy – Final**	**Inclusion of Image Data (Yes/No); Localisation (L) / Regularisation (R)**
Snakes, Rüther et al. [61]	Shifted	Poor	Poor to Mediocre	Yes, snake movement influenced by image forces and geometric constraints; L + R
Snakes, Mayunga et al. [12]	Shifted	Not shown	Mediocre to Good	Yes, snake movement influenced by image forces; L only
Snakes, Žunić & Rosin [86]	Shifted	Mediocre	Good	No, snake movement influenced by rectilinearity measure; R only
Snakes, Mayer [87]	Subset, New, Shifted	Mediocre	Good	Two-phase snake deformation. First phase: No; R only. Second phase: Yes, image forces included; L + R
MDL, Brunn et al. [110]	Subset, Shifted	Mediocre	Good	No; R only
Boundary Section Replacement, Liow & Pavlidis [49]	Subset, New	Mediocre	Good	Image data included via incorporating extracted straight lines; Predominantly R
Corner Detection, Caelli, Stassopoulou et al. [111, 107]	Subset	Mediocre	Poor to Mediocre	No; R only
Rectangular Decomposition, Gerke et al. [112]	New	Mediocre	Good	No; R only

Table 3.2: Summary of boundary modification techniques II.

3.6 Conclusion

Jaynes et al. [42] note that as building detection systems have evolved the trend is towards greater generality in many dimensions: nadir views to general oblique views, single image systems to multi-image systems, and 2D object models in the image domain to 3D models in the world domain. To add to this, it is possible to see an evolution to greater generality in terms of scene content. Early systems focused on scenes of low density formal housing and industrial buildings. Over time, new systems have appeared which aim to interpret images of city blocks, suburbia and importantly, for this work, informal settlements. These scenes differ in important ways from one another and it is reasonable to assume that a tailor-made extraction strategy will be more successful than a generic one. Scale-space and scale-space like approaches have recently become more popular, partly through the availability of commercial tools like eCognition, and these approaches are starting to be applied to the domain of building detection.

The building detection and delineation systems which have been reviewed are mostly structural in nature. Adopting a structural approach enables these systems to identify subparts of the building model and group these to form complete hypotheses. This is most obvious in edge-based strategies in which buildings are usually viewed as being composed of straight line primitives. These primitives are grouped together into intermediate structures which are finally grouped to form parallelograms or aggregations of these. In region-based strategies, uniform regions which correspond to building components (for example roof panels) form the primitives. These regions often exist as part of a hierarchy and they may need to be merged in order to form complete building hypotheses. Integrated strategies, strategies which use height data and scale-space strategies, also utilise these fundamental primitives and deal with similar issues.

It is evident from the literature that for single-image systems edge-based strategies predominate. This is possibly because they are able to exploit domain-specific knowledge of building shape more readily than region-based methods. However, systems which utilise DSMs lend themselves naturally to formation of regions from elevation blobs, and it is interesting to see that scale-space type systems largely utilise region primitives even though this is not a necessity.

A commonality present in the single-image systems reviewed is the use of low-level techniques (edge detection, segmentation or thresholding) which do not rely on domain-specific knowledge, followed by higher-level interpretation steps which

employ domain knowledge (such as the use of shape and shadows for forming and verifying hypotheses). The initial low-level image processing methods tend to perform poorly. This is because building cues are partially or completely lost when the real object is projected into the image space [29]. Additionally, low-level techniques are implemented through the use of local operators which essentially "view" the image through a small aperture and thus have no notion of the scale of the structures being detected. These techniques are therefore unable to discriminate between useful detail at the scale of interest and irrelevant detail, with both being extracted equally well. The remainder of the interpretation process is devoted to filtering, grouping, correcting, and refining the results of these low-level techniques.

If height data is available, altimetric thresholding can be used in bringing domain-specific knowledge to bear early on in the detection process, as shown in some of the automated systems for shack detection. This can greatly simplify building detection as follow-on stages need only disambiguate between buildings and other 3D structures of comparable height such as trees. Having height data available, however, does not necessarily simplify the task of delineation if the data is not that accurate. For example, in [61, 79] it is clear that the elevation blob boundaries do not match up well with the actual shack boundaries and closely-spaced shacks are extracted as aggregated blobs. Automatically generated DSMs from stereo-matching have inaccuracies, especially at surface discontinuities such as shack roof edges [61]. These inaccuracies are attributable to insufficient ground sampling data and matching errors caused by lack of sufficient texture on roofs, poor image quality, occlusions and the presence of shadows [52, 12].

Aside from the above-mentioned broad commonalities, it is evident that there is no standard methodology for building recognition. Most of the different approaches discussed have been developed in an ad-hoc fashion with past experience, intuition and domain-specific knowledge playing an important role both in the choice of techniques to apply and in the choice of parameters for those techniques, as stated in [16].

Shadows play an important role in building and shack detection. They form an invaluable source of scene-domain knowledge by providing cues for both building verification and the estimation of building height [45]. For single nadir-view images shadow is the only height cue. However, shadow is also shown to be useful in systems which do utilise height data [79, 13]. In these systems, shadow is not used for shack verification but instead to identify regions in the image in which shacks do not occur [79] and to provide a reduced set of reliable contours (when compared to the contours

extracted from the source image) for semi-automated shack extraction [13].

Boundary modification techniques are a key part of region-based systems and the main issues involved are digitisation noise removal, localisation and regularisation of the boundary. A variety of techniques for boundary localisation and regularisation have been used with snakes being a popular choice. Snakes offer an elegant way of simultaneously combining the concerns of localisation through the use of image forces and regularisation through the application of domain-specific geometric constraints. However, the literature does not convincingly demonstrate that this combination is effective. Some researchers have used snakes for either regularisation or localisation but not both together [86, 12]. In cases where these aspects are combined the results are mediocre [61] or multiple phases of boundary modification are involved with only one of the phases combining these aspects [87].

Localisation represents a data-driven approach to boundary modification while regularisation represents a model-driven approach. The different concerns addressed by localisation and regularisation relate to the trade-off that is being made between the data informing the object model and the model being imposed on the data. Techniques which do not include image data in the modification process are biased towards model conformance.

This chapter has provided a review of specific building detection systems and strategies which are relevant to the task at hand — the detection and delineation of shacks in informal settlements from single nadir-view image. The following chapter describes the problem in more detail and provides a motivation for using a scale-space.

Chapter 4

The Problem of Shack Detection in Informal Settlements

In this chapter roof substructure is identified as a key factor which makes automated shack detection difficult. It is noted that existing shack detection systems mainly utilise image-generated DSMs to try and overcome this issue. Although strategies based on these DSMs are viable, they have their shortcomings and detection rates could be improved. The investigation of an alternative approach to handling the problem, one which uses a scale-space generated from a single source image is defined as the objective of this thesis. Several reasons are given as to why this is a worthwhile investigation. The scope of the investigation and the research methodology adopted are described.

4.1 Introduction

When performing automated shack or building detection there is a conundrum in that, on the one hand, there is too much distracting fine-scale detail contained in the image, while, on the other hand, there is too little relevant detail as the boundaries of shacks and buildings are not always clearly defined due to low contrast. This chapter examines the former issue in more detail by considering the problems caused by roof substructure, particularly in informal settlement scenes. An examination is conducted as to how existing systems handle the problem of substructure. The use of a scale-space as a potential way of alleviating this problem is presented. The objective of this thesis is then explicitly stated and motivated. A final section discusses the research methodology used.

4.2 Roof Substructure - A Particular Issue in Informal Settlement Scenes

One of the notable characteristics of South African informal settlement scenes is that shack roofs are constructed from a range of diverse materials[1] including plastic, iron sheeting, and timber [6]. These materials, as well as rocks and tyres and other items which may be placed on the top of shack roofs to secure them, are termed *roof substructure*. By the use of the term "substructure", it is to be understood that the scale of objects considered to be substructure are smaller than the scale of the objects of which they are a part. In low resolution images, substructure may not be visible. In medium- to high-resolution images, however, roof substructure is more observable, especially if it contrasts strongly with neighbouring roof areas.

In this work, the goal of the system is to delineate the boundaries of shack roofs. Given such a goal it can be argued that informal settlement scenes contain many roof substructure details that are, at once, irrelevant to the final object model and disadvantageous to the detection process.

Shack roof substructure tends to be variable in nature and is often less observable than the shack of which it is a part, due to its smaller scale. For these reasons, it cannot be usefully employed as a building cue. Moreover, the presence of substructure tangibly affects the appearance of shacks as it causes rooftops in the image to appear inhomogeneous, which results in edges or regions on the interior of roofs being detected by low-level operators. These unnecessary details may be locally significant (due to their high contrast) when compared to the roof boundaries or the average intensity of the roof area. This hampers shack detection as it results in the shack boundary or roof area being highly fragmented at a primitive level. It then becomes difficult to find primitive groupings which correctly delineate the entire shack roof as opposed to part of it, without the presence of other cues.

Although roof substructure is particularly problematic in scenes of informal settlements, it is also documented as being an issue for building extraction systems operating on high-resolution images of urban [37] and suburban [82] scenes.

Areas which surround rooftops may also contain smaller scale structures, such as bushes and cars, which disturb what would otherwise be homogeneous surroundings. This too, complicates detection as it makes it more difficult to separate a shack from

[1] This characteristic is not unique to South African informal settlements but it is particularly marked and widespread in South Africa.

its surroundings.

4.3 How Existing Systems Deal with Substructure

For single intensity images a data-driven approach is initially adopted as image primitives are extracted and grouped according to the object model. If these primitive groupings offer enough evidence to establish an instance of the object model, then the switch to a model-driven approach can take place, as in the hypothesise-and-verify paradigm. The model is now able to tightly constrain the search for further evidence for the hypothesis. It is the initial data-driven or bottom-up process which suffers most from the excess detail present in the image. This excess detail makes it more difficult to formulate building hypotheses and may contribute as negative evidence against hypotheses. However, once an instance of the object model is established, substructure detail can be suppressed by virtue of the fact that it does not fit the model. Additionally, the verification stage is less disturbed by substructure as the search for verification evidence is far more focused, both in terms of the image area being searched, and in terms of the type of evidence being searched for.

Existing edge-based systems which deal with scenes of formal buildings demonstrate that it is possible to discriminate between extraneous detail and genuine building boundary evidence through the extraction of relatively long straight lines. Using geometric constraints, these salient lines can be grouped to form partial or complete hypotheses. From this point on, the search space for further building evidence is greatly restricted and fine scale detail can be largely ignored.

In region-based approaches, roof substructure and other fine detail may disturb what would otherwise be a largely homogenous rooftop, resulting in over-segmentation. Existing approaches make use of knowledge of building shape for guiding region mergers where over-segmentation occurs. Compact substructure which lies unambiguously within verified roof regions can simply be ignored when roof boundaries are to be extracted [49]. If region features are to be calculated based on pixel intensities, then simple averaging can be used [37]. Integrated approaches which use edges to *a priori* determine the extent of regions can apply statistical methods to remove outliers (in terms of colour or intensity) corresponding to disturbances [82].

It appears that the above approaches for dealing with roof substructure, and fine-scale detail in general, are less viable for scenes of informal settlements. This is mostly because of the fact that substructure is so prevalent in these scenes. In [13], for

example, it is clearly shown that the use of attributed contours [82] results in many contours on the interior of shack roofs and fragmented contours along shack borders. This illustrates that the problem is difficult to solve using typical edge-based methods. Improvements are evident in [5] where a specialised edge detector is used, but the problems still occur to a lesser degree and corners appear slightly rounded. In [115] a region growing operator based on a homogeneity criterion (described in Section 5.2.4) is used to identify shacks in an informal settlement scene. The results are poor with many shacks being partially or completely undetected due to their inhomogeneous appearance.

It is unsurprising then, to see that most automated and semi-automated systems for shack detection have made use of height data (see Section 3.3). Roof substructure is far less of a disturbance when locating shacks with the aid of height data such as a DSM. This is because:

- the presence of 3D objects on a roof surface does not adversely affect a technique such as altimetric thresholding which is used to separate ground regions from above-ground regions;
- the use of different materials for roof construction will not impact on the height map, provided these materials do not deviate significantly from the roof plane.

DSM data generated from stereo matching is used in identifying shack instances; shack boundary delineation then takes place, in some instances, in conjunction with image data. However, this is not a panacea because the quality of DSM data produced by stereo matching is insufficient for accurately generating hypothesis boundaries while the presence of fine-scale detail renders image data less useful. These systems are capable of shack detection/delineation but there is room for a fair amount of improvement in terms of quality and accuracy.

To sum up, in by far the majority of single-image building detection systems, locally significant substructure detail is initially extracted from the source image and then filtered and suppressed based on domain knowledge and the object model. Shack detection systems have made use of DSM height data to help overcome problems due to substructure. In all of these systems no attempt is made to modify the image data (pixel intensities) in a way which will simplify the image to remove substructure and fine-scale detail. Scale-space strategies (Section 3.4.4), which do attempt to do this, are small in number and, none, to the knowledge of the author, have been applied to the problem of shack detection.

4.4 Potential Benefits of Using a Scale-Space

Suetens et al. [116] identify the complexity of the source data as roughly corresponding to the amount of semantic ambiguity involved in interpreting the image itself or its higher-level representations. Data which contains many false, incomplete or conflicting instances of the object model, or associated cues, is more complex to deal with than data in which the model instances appear unambiguously.

Using a scale-space for simplifying the image data results in a family of images in which the level of detail varies according to the scale. As the original image is increasingly smoothed, the information content is reduced as edges and regions are eliminated. If these happen to represent substructure then this substructure is annihilated over scale. As *meaningful* information in the image (substructure) is destroyed, the objects of which the substructure forms a part are emphasised[2]. A diffused image, in some sense, can be seen to be an abstracted version of the original where abstraction is defined as an "increase of the degree of simplification and emphasis"[89]. Utilising abstracted versions of the original image offers the promise of better detection rates as the appearance of shacks will be emphasised at certain scales and more readily recognised. In other words, the saliency of relevant detail will be increased, allowing hypotheses to be more accurately delineated or inferred in cases where boundary detail is missing.

A caveat to this is that there is a minimum data complexity which is required to solve the problem [17]. Over-simplification of the image data increases the difficulty of the detection task and ultimately results in the problem not being solvable. Figure 4.1 illustrates this idea.

4.5 Problem Statement and Investigation Scope

The objective of this thesis is to investigate whether a scale-space, in which data is simplified *at the image level*, can be used to facilitate the task of automatically detecting shacks in informal settlement imagery. Shack detection is to be understood as 2D shack roof delineation, as in existing systems.

The use of a scale-space for shack detection is limited, here, to nadir scenes of typical South African informal settlements and takes advantage of a number of assumptions

[2]Noise may also be eliminated in this fashion but it is assumed that the images being dealt with have low levels of noise.

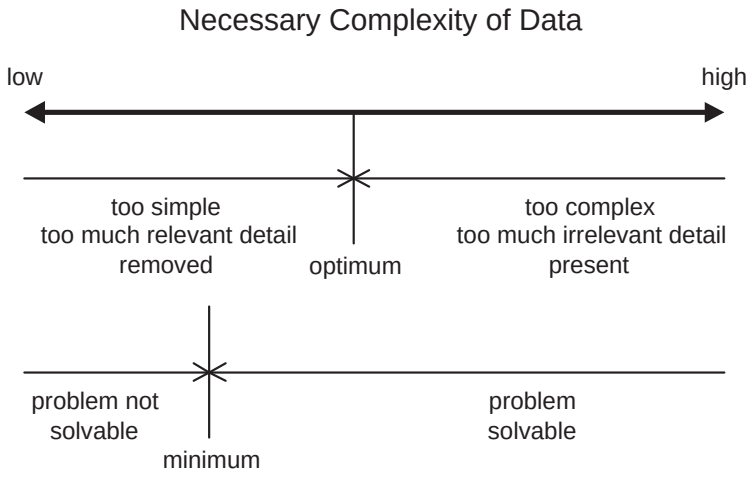


Figure 4.1: Necessary complexity of data (adapted from [17]).

which can be reasonably expected to apply to such scenes.

The following characteristics with regard to the content of informal settlement scenes have been identified in [6] and are assumed to hold:

- The majority of buildings are shacks which are 4-sided, single-storied structures with flat or near horizontal roofs. In other words, both the shack structure and roof architecture can be classified as simple.
- Shacks have fairly rectangular ground plans although deviations from rectangularity of up to 30° are not uncommon.
- Shack dimensions start at around 4 m by 4 m with roof heights of 2–2.5 m.
- Shack roofs are constructed from diverse materials including iron sheeting and wooden planks. Naturally, these materials have different colours and textures. Additionally, rocks, tyres and other objects may be present on top of roof materials.
- Shacks are generally closely clustered with around 2–3 m separation. In some informal settlements (Alexandra, for instance, which is located near the city of Johannesburg) the clustering may be even more dense — to the extent that even a human photo-interpreter has difficulty distinguishing where one shack ends and another begins. Scenes of such highly dense informal settlements are excluded from this study.
- A few formal buildings may be present (for example, schools and community centres) and these tend to be large in size relative to the surrounding shacks. They may also exhibit a slightly more complex roof architecture, for example, an A-frame roof.

- There is relatively little vegetation. Here, it is also assumed that the terrain is fairly flat and images may or may not be ortho-rectified.
- The scene, in general, is fairly unstructured. Shacks are not strictly aligned with one another or with roads.

In terms of object observability (2.4.2), the investigation is constrained by the following:

- The detection strategy targets images of medium resolution (ground pixels sizes between 0.2–1 m). According to [6] the minimum ground pixel size capable of providing useful cues for informal settlement mapping is of the order of 0.5 m and a finer resolution is needed for mapping communal toilet facilities etc.
- The source imagery used is monocular and greyscale (although in some cases these images have been derived from datasets having colour, overlapping imagery of the particular scene).
- The detection strategy is not reliant on height data.
- Images are assumed to be of good quality (reasonable contrast, low sensor noise and so on), as would be obtained from an aerial mapping survey. It is assumed that the images are acquired when strong shadow is present. This investigation does not take into account conditions of adverse observability.

The aim of this work is to produce a system capable of detecting shacks in aerial photographs. For this information to be directly usable in some end-user applications the photographs would need to be ortho-rectified, geo-referenced and so on. However, the focus of this thesis is not on the photogrammetrical aspects of building detection but on the computer vision aspects. The nadir-view imagery to which the system is applied may or may not be rectified.

Finally, this work focuses primarily on investigating and developing mid-level processes which take advantage of the scale-space. High-level processes and the explicit representation of knowledge pertinent to informal settlement scenes are not considered in any depth.

4.6 Motivation for the Investigation

This investigation is worth conducting as interpreted imagery of informal settlements is useful for a variety of end-user goals including settlement monitoring, informal

settlement upgrading and so on. Previous research shows a substantial decrease in the time taken to extract all of the shacks within a scene in semi-automated systems relative to manual ones. It is presumed that increasing the level of automation and decreasing the amount of user intervention will result in even greater time gains.

Ideally, a shack detection system would be fully automated and highly accurate but this does not seem possible without dense and precise height data, and even then it may be difficult to achieve completely accurate results. It is necessary to balance the benefits of accurate automation against the costs of obtaining the additional data that would be required. The use of a scale-space approach does not require higher quality source data or an increased quantity of source data. If a scale-space based system is capable of improving the current state of automation, using existing data, it would offer a distinct advantage.

Furthermore, it is worthwhile automating the detection of shacks from a single optical image. Current systems which utilise height data in the detection process require multiple overlapping images of the settlement in order to generate a DSM through stereoscopic image matching [61, 13, 79]. The images which are acquired need to be automatically matched through the use of custom software. These factors potentially add to the cost of image acquisition and pre-processing, and increase the complexity of the detection system for users.

Alternatively, if a DSM is to be generated, features obtained from the individually interpreting stereo images can be used in feature-based stereo matching schemes, as in [37, 54] and others. A monocular interpretation could also be used to generate regions of interest, and thereby focus the attention of more sophisticated shack detection and reconstruction modules.

Although the scale-space approach presented here is based on the assumption that the source data consists of single image, additional data is often available such as height data produced by stereoscopic image matching, or multi-spectral data if an appropriate sensor is used. This approach is not intended to supplant approaches which make use of richer source data, rather it should be seen as complementary to these approaches, being an attempt to make the best possible use of the optical image component of the source data. Opportunities for integrating this strategy with strategies using DSMs or for fusing the results of this technique with others then become possible. Integrated approaches allow for more successful interpretations of a scene [35, 46], and the need for multi-cue algorithms (with each algorithm providing information which is integrated into the overall interpretation of the scene) within

the context of the shack detection problem has been identified [5].

4.7 Methodology

A conceptual approach in which shack detection is performed within the context of a scale-space is described in Chapter 5. This is supported by a practical implementation of a prototype system with the aim of both validating and identifying shortcomings in the approach with regard to building detection performance. In order to investigate the performance of the system under varying conditions, a number of case studies are conducted with both qualitative and quantitative results being shown. Optimisation of the prototype in terms of computational resources (speed and memory) is a separate issue that is not dealt with here.

In order to answer the research question both the absolute performance of the prototype is important as well as its performance relative to existing approaches which do not make use of a scale-space. Absolute performance indicates the degree to which the system is able to perform automated shack detection. Relative performance determines whether a scale-space approach offers any advantages over existing approaches. This is difficult to answer directly in that current automated systems rely on an image-generated DSM, whereas this system does not — the requirements on source data are different. However, a comparative measure does give some indication as to the relative strengths and weaknesses of this approach.

4.8 Conclusion

Current shack detection systems are unable to fully automate the detection and accurate delineation of shacks in high-resolution aerial imagery. Existing systems tend to be semi-automated, requiring the user to manually identify shacks, or automated with weaker results (a significant percentage of shacks are not detected and/or shacks are poorly delineated). The shack detection problem is difficult mainly due to the disparate roof materials used in shack roof construction, but the weak constraints on roof geometry and the dense clustering of shacks also play a role. Rooftop heterogeneity presents extraneous detail which disturbs the appearance of shacks and hampers the extraction process. It is worthwhile investigating the use of a scale-space as a means of reducing the amount of extraneous detail thus emphasising the appearance of shacks and enabling them to be detected more easily.

The following chapter details a novel detection strategy based on an anisotropic scale-space. The entire strategy is described from the generation of the scale-space to the production of hypotheses and the refinement of their boundaries.

Chapter 5

Shack Detection Strategy

The building detection strategy which has been developed is described in detail in this chapter. Three main processes of the strategy are identified — anisotropic scale-space construction and region extraction, hypothesis verification and selection, and hypothesis boundary expansion — and these each consist of a number of individual stages. The extraction of hypotheses at each scale using a homogeneous region operator, and their verification using an image sampling technique for detecting expected shadow, is demonstrated. The discrete curve evolution technique is presented and used for simplifying the polygons representing hypotheses' boundaries. This enables the calculation of rectilinearity and compactness shape measures. An algorithm utilising these measures is given for regularising boundaries in accordance with the shack model. Overlapping hypotheses from different scales are seen to be in competition and a fuzzy rule system is presented which selects the best hypothesis from each competing set. Techniques for grouping hypotheses, and for improving their localisation by incorporating edge information, are discussed.

5.1 Introduction

This section provides an overview of the detection strategy that has been implemented. This is done by outlining the conceptual thinking behind the chosen approach and providing links to subsequent sections which delve into the details of the various processes involved.

One of the central ideas of this thesis is that the appearance of the shacks in

the original image may not be optimal for detection. This is because the shack roof substructure disturbs what would otherwise appear as a uniform roof. These substructure details, such as differences in roof materials, negatively impact detection strategies, in that, in the early stages of detection many primitives are found which correspond to substructure as opposed to the shack's entire roof. This complicates follow-on processes, such as grouping, which are now required to differentiate between substructure and structure details.

What is desired is a method of transforming the source image so as to discard irrelevant detail. This is achieved through the construction of anisotropic scale-space which consists of multiple, blurred or simplified versions of the original image. Blurring removes substructure details and in many cases improves the appearance of shacks for detection by making their rooftops appear more uniform. Homogeneous regions are then extracted from each image in image stack or scale-space as it is hoped that at a particular scale a homogeneous region will correspond to an entire shack roof. The creation of the image stack from a single source image, homogeneous region extraction, and region linking across scales, together, form the first of three main processes in the detection strategy and are discussed in Section 5.2. The next process in the sequence is the verification and selection of the regions/hypotheses that have been extracted. Verification involves reducing the set of all regions that have been extracted to those which are taken to be confirmed shack hypotheses. This process is fairly involved and consists of a number of stages, including converting the pixel-based representation of the homogeneous region boundaries to points on a 2D-plane, digitisation noise removal, hypothesis verification, and model-driven boundary simplification. Hypotheses are then selected across scales and grouped. The details are to be found in Sections 5.3 through 5.7.

The final process is one of expanding the boundaries of confirmed hypotheses using straight line approximations to edges which have been detected in the original image. This is presented in Section 5.8. Figure 5.1 illustrates all three processes, their inputs and outputs, and how they are related to one another.

A walk-through of the entire system is given in this chapter. The intermediate results of each detection stage are shown for the 'Marconi Beam 1' image (Figure 5.2), which forms part of the test dataset. This image has been extracted from an aerial dataset of Marconi Beam [83] and converted from RGB colour to greyscale. The ground pixel resolution of this image is 0.18 m. Note that Chapter 6 and Appendix A present the final results from additional images.

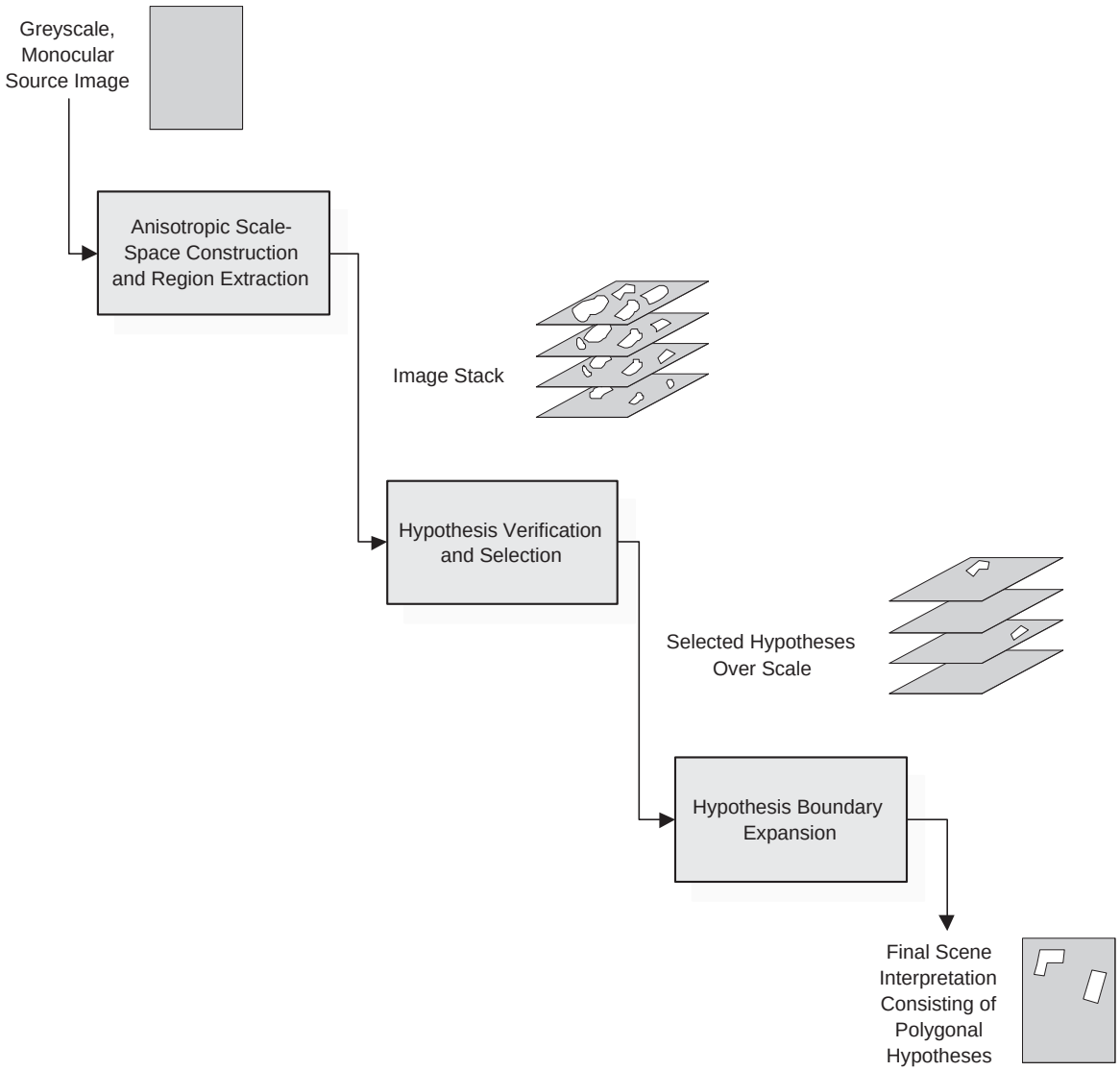


Figure 5.1: An overview of the shack detection strategy.

5.1.1 Manual Input

The automated processes outlined above depend on a number of manual inputs:

Manual identification of both a small and large shack/building

This involves delineating the outlines of two shacks/buildings in a point-and-click fashion. The reason that the system requires this input is so that a range of valid building sizes can be established. This enables extracted regions (Section 5.2.4) that are either too small or too large to be buildings to be removed which reduces the overall processing time as no verification of these regions is required in latter stages. Note that the manually generated outlines

Figure 5.2: Marconi Beam 1 - greyscale source image.

of these two structures are discarded — all the hypotheses shown in both the intermediate results and final scene interpretation are system generated.

Specification of the shadow threshold

Shadows are normally darker than most objects in a scene and therefore can be relatively well detected in scenes with good contrast, by regarding all pixels below a certain intensity level as shadow. These pixels lie in the lower portion of an intensity histogram of the image. Manually specifying the threshold and viewing the results allows the user to gauge how well shadows are being identified. This is particularly important for lower contrast images where the shadow threshold can cleanly separate shadow areas from other dark areas in the image.

Informal settlement scenes may contain clusters of shacks which are closely spaced. In these parts of the scene, the majority of a shack's shadow may fall against the wall of neighbouring shacks with the result that very little shadow is visible in the narrow spaces between the shacks. In order to alleviate this problem, the presence of shadow is exaggerated by morphologically dilating the detected shadow regions with a small disc-shaped structuring element. Figure 5.3 illustrates the shadow found in source image (Figure 5.2) using this

method.

Manual identification of a vertical shadow line

Vertical shadow lines are cast by vertical edges of buildings and, for nadir-view images, are at the same angle as that of the sun vector. Manual identification is performed by the user marking the start and end points of a single shadow line for any shack/building in scene. The length of this line is important because together with the angle valuable knowledge about where to expect shadow in the scene in relation to a hypothesis is gained. For this reason, the user needs to identify a shadow line belonging to a shadow which is not occluded by another structure and which is cast by a shack or building of typical height. In informal settlements most shacks tend to have roof heights of 2–2.5 m [6] so the overriding concern is not structure height but finding an unoccluded shadow.

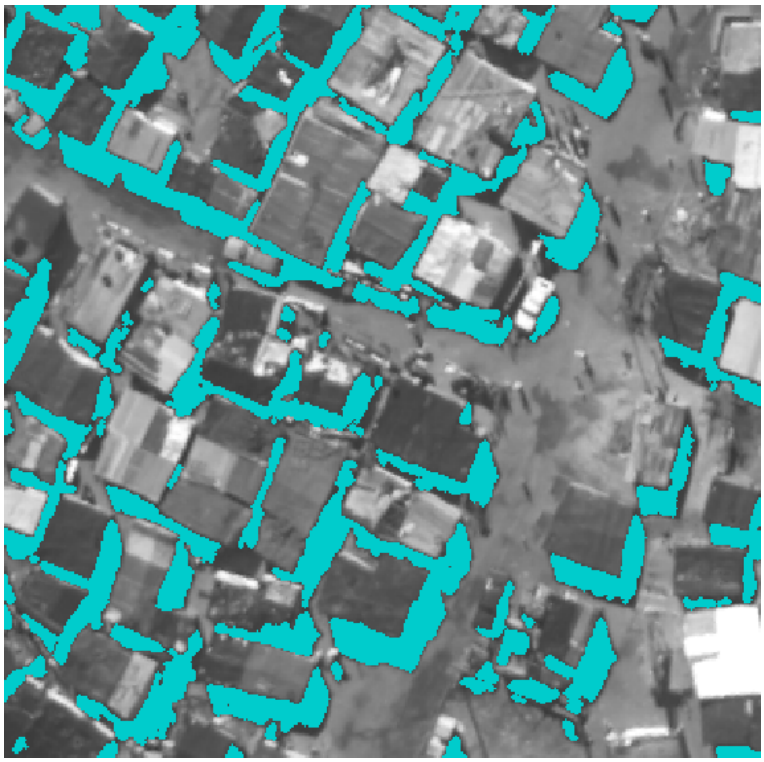


Figure 5.3: Identified shadow (in cyan) from thresholding pixel intensities and dilating the result.

Shadow and sun vector information are used at different stages, primarily for hypothesis verification. Their use is explained in more detail in Section 5.4.

All of the above manual inputs constitute image-specific parameters in that they vary depending on the particular image being processed. These are, in fact, the

only image-specific parameters for the entire system; all other parameters are image-invariant. The full list of system parameters, their meaning and chosen values, is given in Appendix B.

5.2 Anisotropic Scale-Space Construction and Region Extraction

As discussed in Section 3.4.1 there are many different types of scale-spaces. In this work, an anisotropic diffusion scale-space is used as opposed to a linear scale-space. This is because Gaussian blurring rounds object edges and delocalises them over scale. This makes objects at coarser scales increasingly difficult to detect if shape is an important criterion. In this work, building structures are detected at multiple scales partly based on shape rectilinearity — using a non-linear diffusion model supports this in preserving region shape and significant edges as scale increases.

5.2.1 Anisotropic Diffusion — The Perona-Malik Model

"Diffusion" is readily understood as the physical process of establishing equilibrium where differences in concentration exist. For example, consider a drop of food colourant that is placed in a glass of water. Over time the colour will spread throughout the water until eventually the water's colour becomes uniform. The colourant moves from regions of higher concentration to those of lower concentration until equilibrium is achieved.

The partial differential equation which models this sort of behaviour is known as the *diffusion equation*. Weickert [97] presents the general form of this equation as follows:

$$\partial_t u = \mathrm{div}(D \cdot \nabla u) \tag{5.1}$$

where

$\partial_t u$ = change in concentration u with respect to time t
div = divergence operator, and "$\cdot$" = dot product
D = diffusivity tensor or function
∇u = concentration gradient

In image processing the "concentration" should be understood as the image intensity at a particular location. Likewise, the "concentration gradient" is taken to refer

to the image intensity gradient. Finally, time should be interpreted as *scale* as the image will evolve over scale as the diffusion algorithm is iteratively applied.

Restating the diffusion equation (Equation 5.1), using symbols appropriate for the context of image processing and assuming that D is a diffusivity function g producing positive scalar values (the direction of diffusion will be parallel and opposite to that of the concentration gradient), one arrives at the Perona-Malik equation for anisotropic diffusion [117]:

$$\frac{\partial}{\partial t} I(x, y, t) = \operatorname{div}(g(\|\nabla I(x, y, t)\|) \nabla I(x, y, t)) \tag{5.2}$$

where

$I(x, y, t)$ = image intensities of the smoothed 2D image at a particular scale t, and position (x, y)

$\frac{\partial}{\partial t} I(x, y, t)$ = change in image intensities with respect to scale

$\nabla I(x, y, t)$ = image intensity gradient, and $\|\nabla I(x, y, t)\|$ = magnitude of the gradient

$g(\,\cdot\,)$ = diffusivity function, a function of the image gradient magnitude

This equation represents the first published non-linear diffusion filter or model [97]. The solution of equation 5.2 using the original image as the initial condition results in a smoothed image at scale t.

Perona and Malik propose two different functions for g which is called either a conductivity function [117], a diffusivity function [97] or an "edge-stopping" function [118]. The first of these functions is used in generating the image stack, and is as follows:

$$g(\|\nabla I(x, y, t)\|) = e^{-(\|\nabla I(x,y,t)\|/K)^2} \qquad (K > 0) \tag{5.3}$$

where

K = diffusivity constant

Equation 5.3 is a function of the magnitude of the local image gradient at scale (iteration step) t. This effectively means that the amount of diffusivity varies spatially in the image depending on the image gradient in the local neighbourhood. The diffusivity is a monotonically decreasing function which tends to 0 as the magnitude of the image gradient increases. This is shown in Figure 5.4a.

In neighbourhoods where the gradient magnitude is large, that is, close to edges, the

amount of diffusivity is low, hence the term "edge-stopping" function. On the other hand, in regions of fairly homogeneous intensity, the image gradient magnitude is low resulting in a high diffusivity value.

To fully understand the effect that the diffusivity function has within the context of the Perona-Malik equation (5.2) it is worthwhile defining the flux function as in [119]:

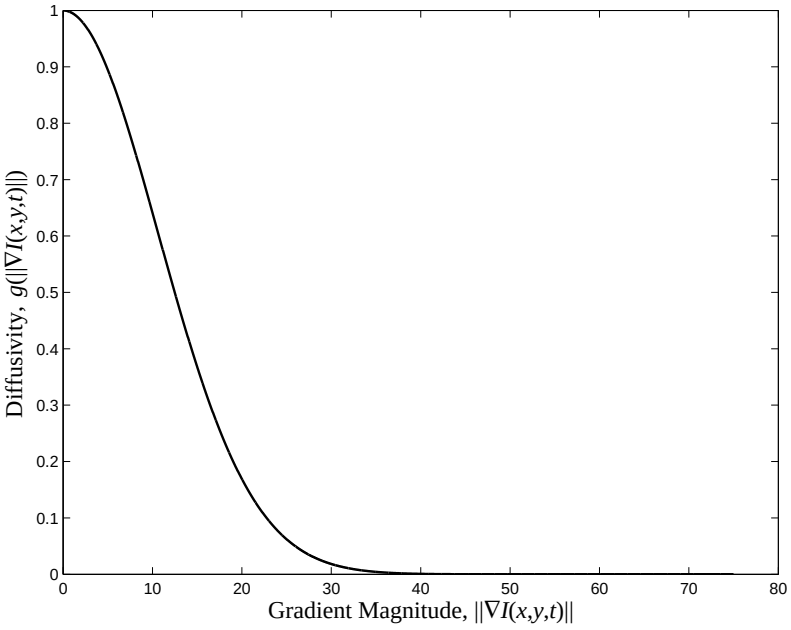
$$\Phi(x, y, t) = g(\|\nabla I(x, y, t)\|)\nabla I(x, y, t) \tag{5.4}$$

which allows Equation 5.2 to be restated as $\frac{\partial}{\partial t} I(x, y, t) = \text{div}(\Phi(x, y, t))$.

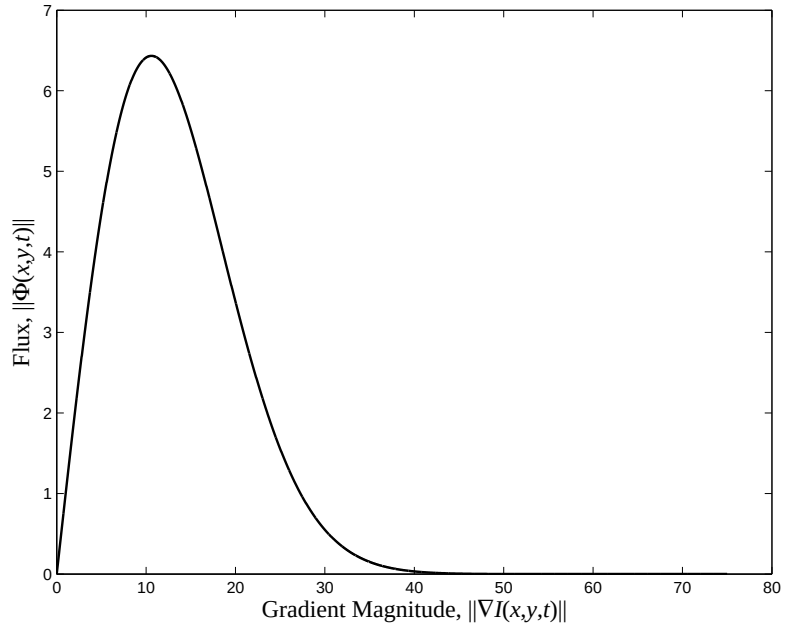
The flux magnitude with respect to the image gradient magnitude, using the diffusivity function given by Equation 5.3, is plotted in Figure 5.4b. Here it can be observed that the flux magnitude increases with image gradient magnitude up to a maximum and then decreases to zero. This means both homogeneous regions and high contrast edges will experience very little smoothing flow and will remain unchanged. Lower contrast edges with gradient magnitudes in the vicinity of the peak flux magnitude will, however, experience a large amount of flow which will cause them to be eliminated over scale. Over many scales flux will decrease in all parts of the image and the image will tend towards a steady state.

The constant K helps to determine the peak of the flux function and hence which gradient magnitudes experience maximum smoothing. If K is set at too large a value then the flux is high for nearly all gradient strengths which results in excessive blurring of all image edges. Ideally, the flux should be low for high contrast edges, if they are to be preserved. Experimentally, it was determined that setting $K = 15$ (refer to Appendix B) provides a sufficient, yet not excessive, amount of smoothing. Decreasing the value of K, increases the height in the stack at which a given strong edge will blur out of existence. Therefore more conservative (slightly lower) values of K are still viable. Increasing the value of K, however, runs the risk of eliminating important edge information from almost all stack levels.

The Perona-Malik equation can be reformulated as two distinct diffusion processes — one across the edge and one perpendicular to it [120, 121]. This reformulation reveals that for high contrast edges (with gradient magnitudes greater than $K/\sqrt{2}$, the point at which the flux function peaks) the diffusion process across the gradient *reverses* as if it were running backward in time (scale) which may result in edges being enhanced. This backward diffusion process is ill-posed and known to be highly sensitive to noise, however, it has been shown that discretisations (Section 5.2.2) of the Perona-Malik equation introduce a strong regularising effect [121].



(a)



(b)

Figure 5.4: Diffusivity function g and flux Φ versus image gradient magnitude for $K = 15$. The peak of the flux function occurs when the gradient magnitude $= K/\sqrt{2}$.

5.2.2 Discretising the Perona-Malik Equation

The above formulation for anisotropic diffusion is for a continuous, two-dimensional image surface. In order to apply this diffusion process to a raster image, it is necessary to use a discrete approximation.

Perona and Malik [117] discretise the anisotropic diffusion equation (5.2) as follows (presented as in [118]):

$$I_s^{t+1} = I_s^t + \frac{\lambda}{|\eta_{s_4}|} \sum_{p \in \eta_{s_4}} g(|\nabla I_{s,p}|) \nabla I_{s,p} \tag{5.5}$$

where

I_s^t = pixel intensity at position s in a discretely-sampled image, at time step or iteration, t

λ = a positive, real constant that determines the rate of diffusion or the degree of influence that the neighbouring pixels have on the intensity of pixel s

η_{s_4} = 4-connected spatial neighbourhood of pixel s

$|\eta_{s_4}|$ = number of pixel neighbours, which is always four, assuming that the image boundaries are padded

The image gradient from the pixel s to each of its four nearest neighbours (North, East, West and South) is approximated using finite differences:

$$\nabla I_{s,p} = I_p - I_s^t, \quad p \in \eta_s \tag{5.6}$$

where

I_p = pixel intensity of a 4-connected neighbour of a pixel at position s

The diffusivity, $g(|\nabla I_{s,p}|)$, is calculated using equation 5.3.

5.2.3 The Image Stack

λ in Equation 5.5 can be set to a maximum value of 0.25 without sacrificing numerical stability [117]. This reduces the number of parameters that need to be specified for creating an image in the anisotropic scale-space to two: the constant K in the diffusivity function and the number of iterations t required to produce the image. As mentioned earlier, K is set to 15. Table 5.1 gives the number of iterations that

are used in generating each image in the scale-space or image stack. In total, nine images form the image stack with the first image in the stack being the original image. The number of iterations for each stack level have been chosen heuristically. This is explained in more detail in Section 5.2.5.

Stack Level	1	2	3	4	5	6	7	8	9
Number of Iterations	0	2	3	5	10	15	20	30	80

Table 5.1: Image stack levels and number of iterations. Stack level 1 is the original image.

Figure 5.5 displays selected images from the image stack. As the stack level or scale increases, the image becomes progressively smoothed or abstracted as fine details are removed. This process makes the shack roofs appear more uniform which aids in their detection. However, it is also noticeable that too much blurring can result in sections of a shack roof's boundary being eliminated and the boundary along these sections becomes indistinguishable from its surroundings. Finally, it is worth noting that different shack roofs appear best (uniform and generally distinguishable from the surroundings) at different scales.

5.2.4 Using the Homogeneous Operator for Region Extraction

The multi-scale representation of the original image does not explicitly contain information regarding the objects of interest. In order to detect these objects over scale it is necessary to apply a feature detector to each of the images in the image stack and integrate the results.

As the appearance of shack roofs becomes more uniform over scale it seems germane to use a feature detector that specialises in identifying homogeneous regions. Such a feature detector, termed a *homogeneous operator*, has been devised. The operator simply calculates the average absolute difference in pixel intensity between every pixel in a square neighbourhood and the centre pixel. This is expressed by Equation 5.7 and the equation given earlier for neighbouring pixel differences (Equation 5.6). In essence, the magnitude of the approximate gradient from the centre pixel to each of its neighbours is averaged.

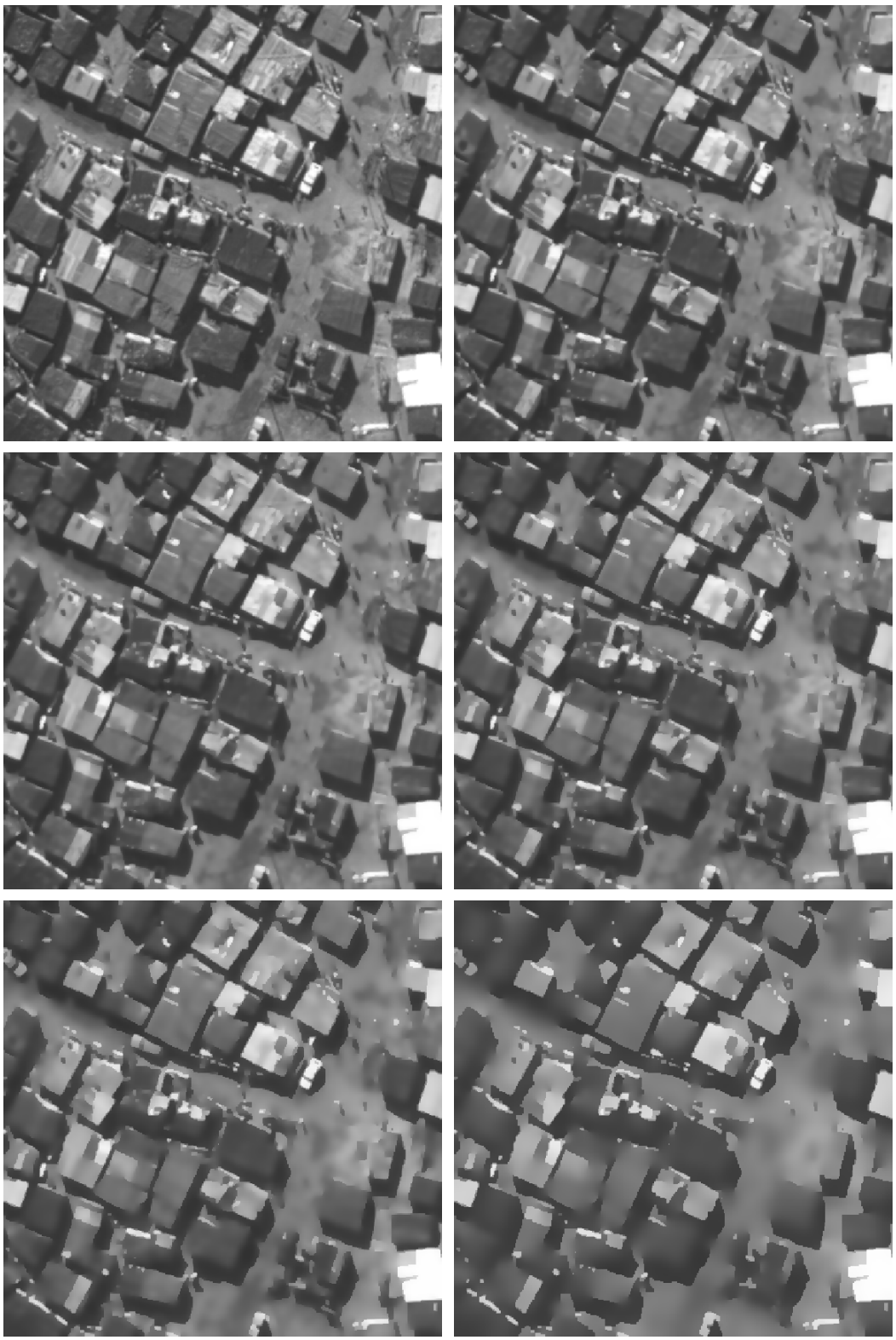


Figure 5.5: Selected images from the anisotropic diffusion scale-space. Original image at top left. From left to right, top to bottom: stack levels 1, 3, 4, 5, 7, and 9 corresponding to 0, 3, 5, 10, 20, and 80 iterations.

$$H_s = \frac{1}{|\eta_{s_8}|} \sum_{p \in \eta_{s_8}} |\nabla I_{s,p}| \tag{5.7}$$

where

H_s = homogeneous operator output at pixel position s

η_{s_8} = 8-connected spatial neighbourhood of pixel s

$|\eta_{s_8}|$ = number of pixel neighbours, which is always eight, assuming that the image boundaries are padded

$\nabla I_{s,p}$ = difference between pixel intensities at positions s and p

The homogeneous operator represents a non-linear, spatially invariant filter. The result of applying this filter to the source image is shown in Figure 5.6. Note that in order to adequately visualise the operator output, the output values, which are in a very narrow range, have been mapped to the entire greyscale range. It can be seen in Figure 5.6 that areas exhibiting high contrast in the original image, such as some roof and shadow boundaries, appear white; uniform areas appear dark due to their high degree of homogeneity. Areas of intermediate contrast appear in grey.

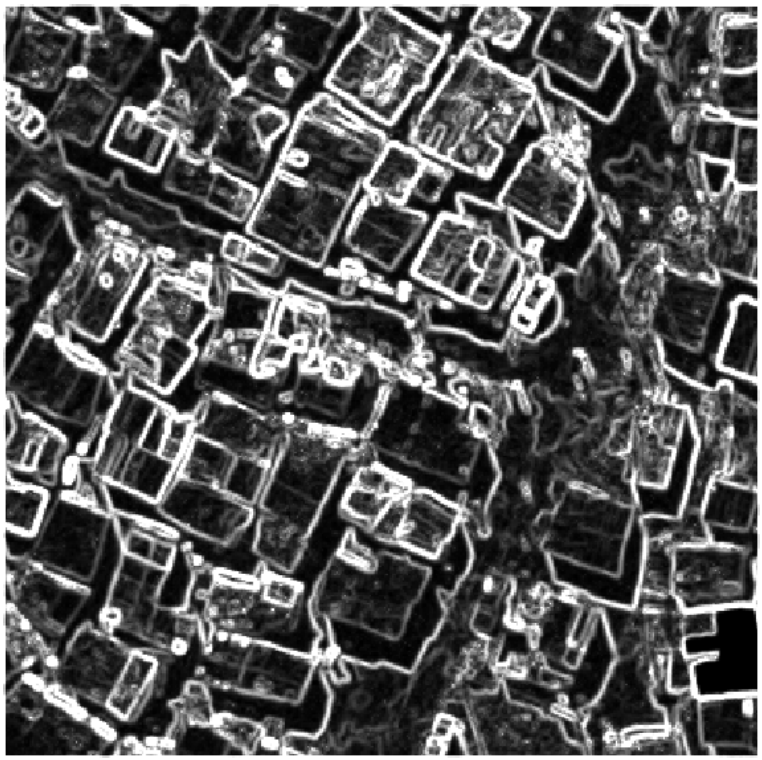


Figure 5.6: Homogeneous operator output for Figure 5.2. The higher the homogeneity value, the darker the pixel colour.

The filtered image illustrates the difficulty of the shack detection problem in that

there is a large amount of locally significant detail on the interior of shack roofs as well as in areas surrounding shacks. Shadows are quite well delineated, as shown in other studies [13, 79, 45].

In order to extract homogeneous regions, the homogeneous operator output is thresholded and all pixels having a value less than the threshold are extracted as connected components. This threshold is the same for all levels in the image stack and it was determined experimentally. The system is not that sensitive to the actual threshold value as long as it is chosen to be "conservative" such that only *highly* homogeneous connected components are identified in the original image (which are unlikely to correspond to entire roofs). As the homogeneity of the image increases over scale, the connected components will increase in size and entire roofs will emerge.

Connected components are morphologically 'opened' using a square 3-by-3 structuring element. Opening removes pixel offshoots and thin pixel bridges that may occur between mostly unconnected regions. In images of informal settlements this is an important step, as shacks are often closely spaced and opening prevents premature mergers between regions corresponding to different roofs.

Finally, all holes are filled in the extracted regions as subsequent stages only rely on the region boundaries. Hole filling plays a valuable role in that compact roof substructure, such as a rock or tyre which sharply contrasts with the surrounding roof interior, and therefore disturbs its homogeneity, is removed. This cannot be accomplished by the anisotropic diffusion filter if the edge magnitudes of such substructure are similar in strength or greater than that of the roof boundaries because the diffusion process equally blurs or preserves all edges of similar magnitude. Figure 5.7 displays the extracted regions for selected images in the image stack.

Table 5.2 gives the number of regions extracted at each stack level. Overall there is an expected downward trend as scale increases; however, it can be noticed that the number of regions increases from stack level two to stack level three. This is because the diffusion operator at fine scales increases the homogeneity of the image, giving birth to new areas detected by the homogeneous operator. As the scale further increases all regions grow in size and mergers start to take place reducing the total number of homogeneous regions.

The final step in this stage is to filter the large number of regions that have been extracted at each scale so that regions corresponding to clearly invalid hypotheses are excluded from further processing. Two criteria are applied:

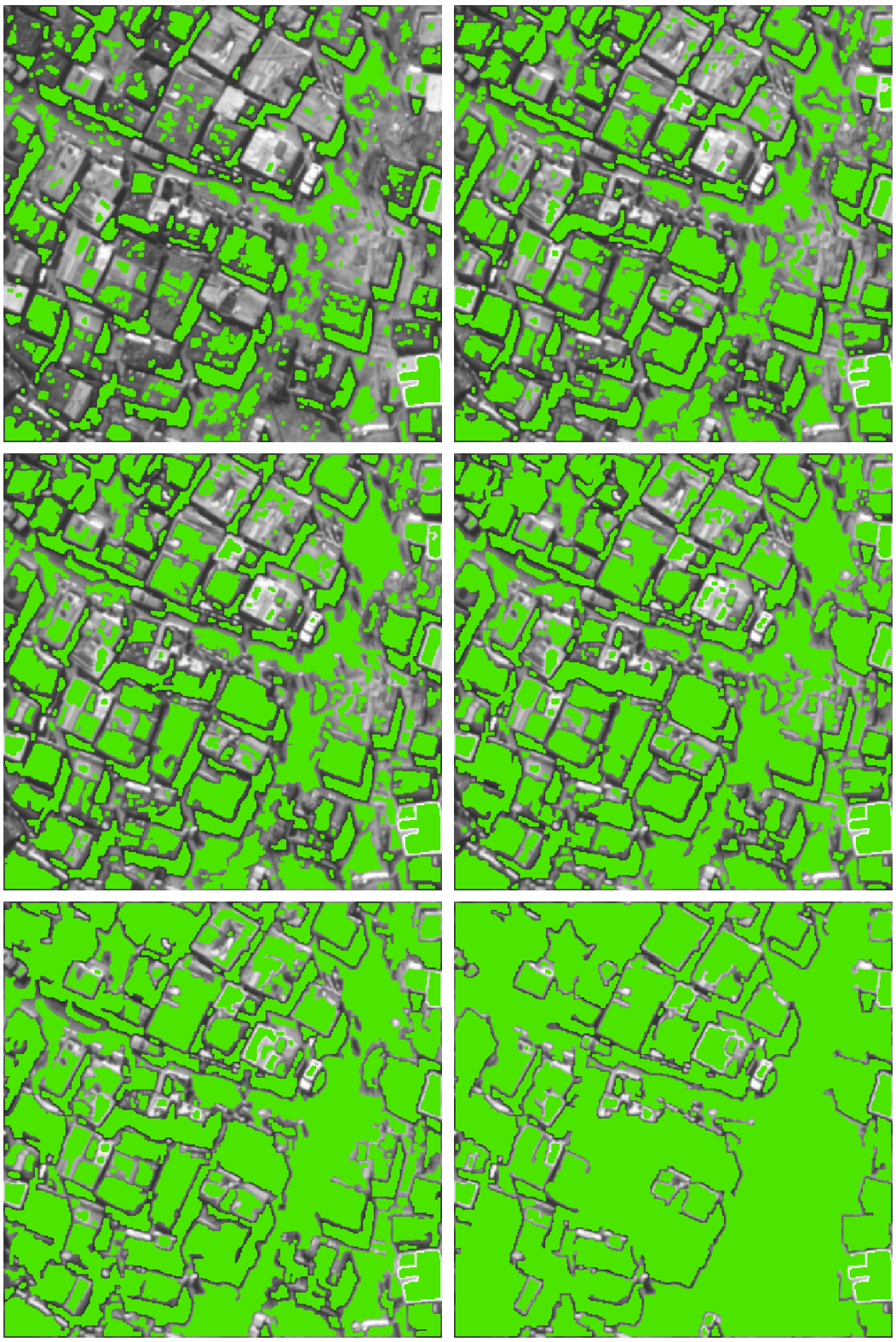


Figure 5.7: Extracted homogeneous regions (in green) from selected images in the image stack. From left to right, top to bottom: stack levels 1, 3, 4, 5, 7, and 9.

Stack Level	1	2	3	4	5	6	7	8	9
Number of Regions	303	269	284	273	246	207	176	126	76

Table 5.2: Image stack levels and number of extracted homogeneous regions.

1. All regions which are either too large or too small to be considered shacks or buildings are removed. The size limits for acceptable regions are based on earlier user input (see Section 5.1.1).
2. All regions which overlap identified shadow (Figure 5.3), by more than a certain extent are removed.

This is illustrated in Figure 5.8.

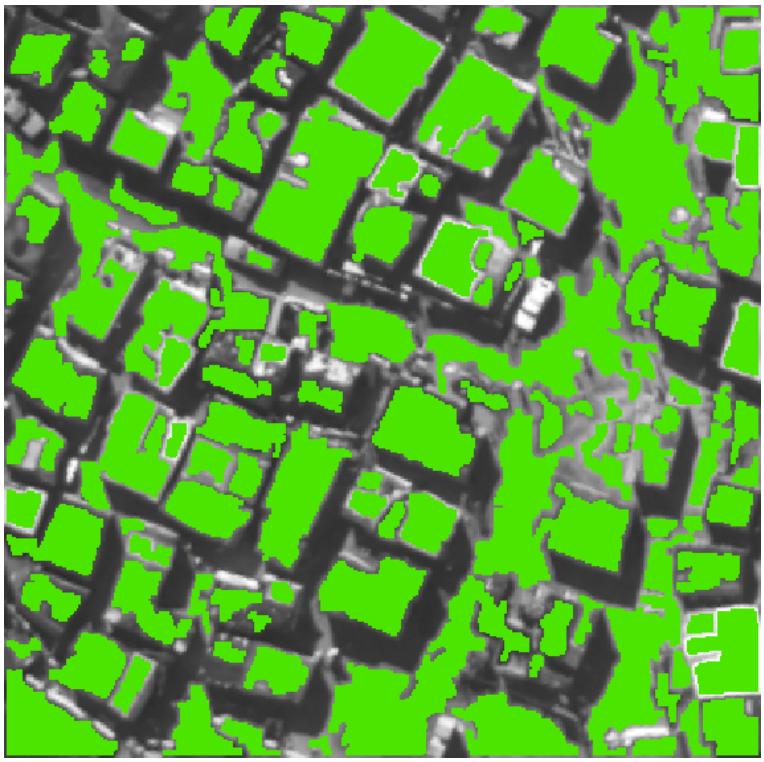


Figure 5.8: Regions remaining after size and shadow-overlap filtering. The remaining regions from *all* stack levels are superimposed on one another in this image.

5.2.5 Optimal Scale for Roof Extraction and Scale-Space Sampling

In order to determine the optimal scale for shack roof extraction it is worth considering the ideal scenario. In this scenario, it is assumed that all possible images on the

scale-space continuum are used for detection, that is, that there is no sampling of the scale-space. Additionally, it is taken for granted that shack roofs are fairly well defined with respect to their surroundings.

If this is the case then the extracted homogeneous regions could be expected to behave over scale in the manner depicted in Figure 5.9. As scale increases, blurring increases and individual regions merge together to form a single region covering the entire shack roof (Figure 5.9 c). This is the optimal scale for detection. In scales prior to optimal scale (Figure 5.9 a and b) blurring is insufficient to create a uniform appearance of the shack roof and the roof consists of a number of distinct homogeneous regions. In scales coarser than the optimal scale (Figure 5.9 d), blurring has increased to such an extent that a part of the shack's boundary no longer exhibits sufficient contrast to separate the shack roof from its surroundings. In these cases flooding of the roof region into the background occurs.

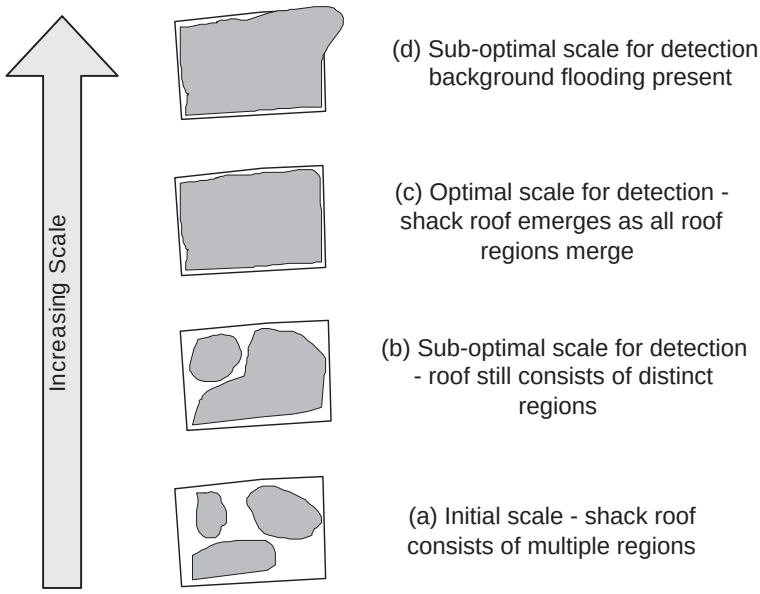


Figure 5.9: Ideal scenario for shack detection.

In practice, the image at the optimal scale for detecting a particular structure may not exist within the image stack for the following two reasons. Firstly, detection takes place using images at selected scales because it is impractical to utilise images at every scale due to the extensive computational processing that would be involved. In other words, the scale-space is *sampled* at particular intervals. None of the resulting images may, in fact, constitute the optimal scale for the detection of a particular shack.

Secondly, and more fundamentally, the ideal scenario makes the assumption that the contrast between different roof materials is less than the contrast between the shack's outer boundary and the surrounding ground. Accordingly, the different roof regions will blur together *prior* to a merger occurring between a roof region and the surroundings. However, this is often not the case and the contrast between the shack roof and the surroundings is less than the contrast between roof materials. In these situations background flooding occurs prior to the emergence of the shack roof. Indeed, in certain cases the contrast between roof materials is so marked that the corresponding regions fail to merge throughout the scales constructed.

In order to resolve the first issue it is essential to adequately sample the scale-space so that one (or more) of the images in the image stack are sufficiently close to the optimal detection scales of the building structures in the scene, to enable recovery of the actual boundaries. In Section 5.2.3, Table 5.1 presents the number of iterations used in producing each image in the image stack. These iterations represent the intervals at which the scale-space is sampled. Experimentally, it has been determined that closely spaced intervals at fine scales, increasing to largely spaced intervals at coarser scales, produces an adequate sampling of the scale-space. Other research into using an anisotropic diffusion scale-space for segmentation has similarly been based on exponential scale sampling [98]. Note, that unlike linear or morphological scale-spaces, the spatial extent of the anisotropic filter does not relate to the scale level and cannot be readily used to enhance the appearance of objects of specific sizes. This makes it difficult to determine the sampling interval theoretically, as is possible with other scale-spaces.

The second issue mentioned above, low figure-to-ground contrast, results in regions being extracted whose boundaries correspond poorly at points to the actual shack boundaries. In order to rectify this, an attempt is made to recover the true shape of each shack boundary from the imperfect region boundary extracted. Techniques for boundary recovery are especially needed within the context of a scale-space approach because the "low figure-to-ground contrast" problem ceases to be limited to specific shacks at a single scale (the original image). Shacks are now extracted at multiple scales and if these are sub-optimal for detection boundaries will be distorted. The boundary recovery techniques used and developed involve digitisation noise removal through discrete curve evolution (Section 5.3.2), model-driven boundary simplification (Section 5.5) and boundary expansion (Section 5.8).

5.2.6 Linking Regions Across Scales

Every homogeneous region extracted at every level is taken to be a candidate building or shack hypothesis. Due to the multi-scale nature of the detection strategy, regions corresponding to the same shack/building appear at multiple scales. Linking these regions together using an overlap criterion makes this explicit. Furthermore, these linked regions are seen to give rise to competing hypotheses which need to be resolved (Section 5.6).

Overlap or the coverage of one region by another can take place in two different directions, depending on whether the region at lower stack level is projected onto the upper level region or vice-versa. This is illustrated in Figure 5.10 for overlapping regions at stack levels n and $n+1$. Two regions are only linked together if the lower region coverage is $> 50\%$ (refer to Figure 5.11). This criterion ensures that a lower region can never be linked to more than one upper region. However, it may result in regions which remain unlinked across scales.

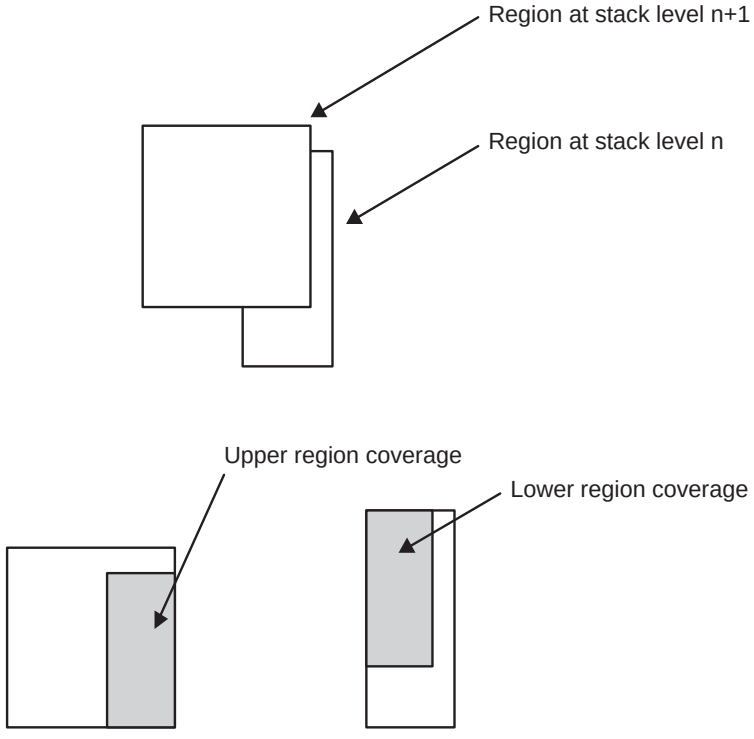


Figure 5.10: Upper and lower region coverage.

This linking process results in a forest of trees[1] in terms of graph theory (see Figure 5.12). Each node or vertex represents an extracted region at a specific stack

[1] A tree is a connected acyclic graph. A forest is a graph in which the components are trees.

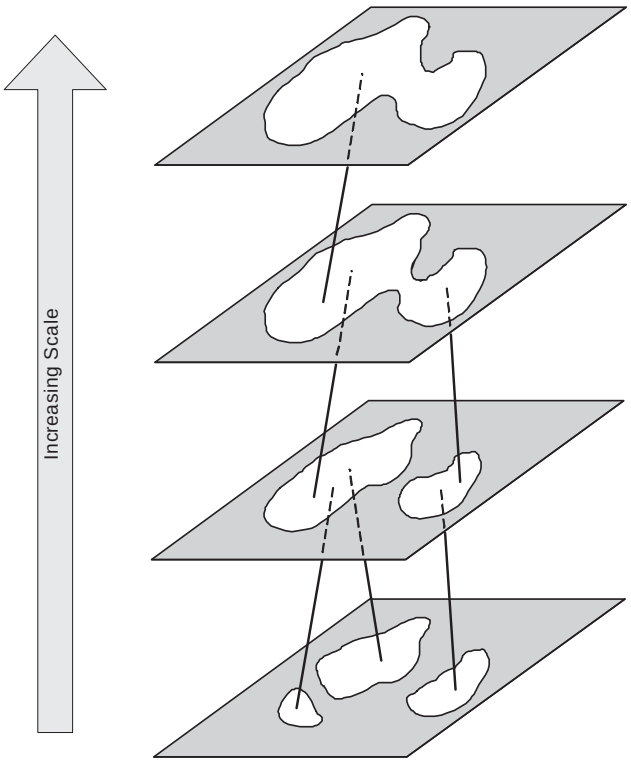


Figure 5.11: Linking regions across scales.

level. The graph edges represent a linkage between two regions at adjacent scales. Multiple regions exist at the highest stack level and each of these forms the root of one of the trees. For the images used in this thesis, most trees span the entire stack height. This is due to the nature of the blurring process which causes extracted regions to increase in area over scale, which in turn results in the overlap criterion being met.

Lindeberg [106] discusses four types of scale-space events that occur with regard to extracting deep structure. Although Lindeberg's approach is based on critical points, the events described are relevant here. These events are:

- *Merge* - a merge event in this context is when two regions form a single region at the next level up in the stack. This is by far the most predominant event.
- *Creation* - creation occurs when a homogeneous region is extracted at a particular stack level and it does not overlap with a region at a lower level. Creations occur because the overall homogeneity of the image is increasing with scale which gives rise to new regions at levels other than the base level.
- *Split* - in principle, a split event occurs when a region at level n is overlapped by two regions at level $n + 1$. It is possible to envisage a region split occurring

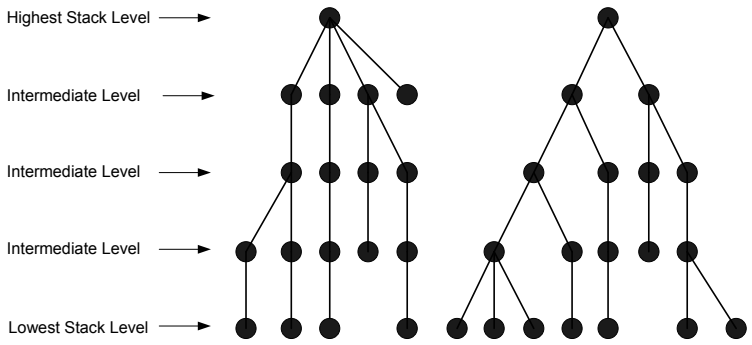


Figure 5.12: Extracted homogeneous regions are linked together across scales to form a forest graph. This example contains two trees. Note that some leaf nodes do not occur at the lowest stack level due to homogeneous regions emerging at other levels.

in the following situation: a homogeneous region is extracted at level n but contains within it a subtle edge, the magnitude of which is insufficient to split the region as all pixels fall below the homogeneity threshold. At level $n+1$ the edge is enhanced due to the anisotropic filter. Pixels corresponding to the edge now exceed the homogeneity threshold and split the region.

Both the homogeneity calculation and anisotropic diffusion depend on image gradient magnitudes approximated by nearest-neighbour differences. Earlier it was mentioned that the homogeneity calculation is essentially the average gradient magnitude of the central pixel to each of its eight neighbours. The total diffusivity for the pixel depends, in part, on its gradient magnitude to its 4-connected neighbours. Given that anisotropic diffusion only enhances edges with magnitudes greater than $K/\sqrt{2}$ (see Figure 5.4), the values of K and the homogeneity threshold can be judiciously chosen to avoid splits. If $K/\sqrt{2}$ is much larger than the homogeneity threshold then edges with the potential to split regions over scale will have already split them at the finest scale. This is the approach adopted.

In practice, the linking scheme presented does not account for split events, so should a split occur it would be indistinguishable from a creation event in the scale-tree.

- *Annihilation* - annihilation occurs when a region is *decreasing* in size as the stack level increases and eventually disappears from one stack level to the next.

Experimentally, it has been observed that these events occur rarely and involve regions which attain very small maximum areas.

As mentioned, the purpose of linking is to allow a correspondence to be formed between regions (hypotheses) at different scales which mask the same image object and are seen as competing. However, linked regions can also be analysed as they evolve (change shape and merge) over scale. This offers the possibility of making more in-depth use of the so-called deep structure of the image, although this is not attempted here.

Finally, it is worth noting that the linking process presented is not as general as others documented in the literature. This is because only extracted homogeneous regions are linked (rather than all image pixels/regions), and the overlap criterion is chosen so that a child region may only have a single parent at a higher scale (that is, split events are ignored).

5.3 Boundary Simplification for Noise Removal

The scale-space construction and region extraction process is based entirely on pixel data. All calculations, described so far, utilise pixel intensities from the source image or derived images. The following processes, on the other hand, primarily rely on a "points on a 2D plane" representation. To convert from a pixel-based representation to a 2D-point representation (a raster to vector conversion), a polygon is derived for each region from the co-ordinates of pixels forming the region's outer boundary. These pixels are 8-connected and therefore the resulting polygon consists of short line segments at the *8-connected angles* of {0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°}.

5.3.1 The Need for Simplification

Simplifying the polygons obtained from vectorisation brings immediate benefits in terms of both processing time and storage space as there are simply fewer data points to deal with. However, the main reason for simplification in this thesis is to remove the noise that the vectorisation process introduces in the form of jagged boundary lines which ideally should be smooth. This noise, if not removed, can severely effect shape measures to the extent that they become unreliable. This is demonstrated for the rectilinearity measure [122] which is used later on (see Section 5.5.1).

The problem of boundary simplification is one of polygonal curve fitting and is often framed as follows: find an approximating polygon, Q, for a given polygon, P, that has the minimum number of sides within a given error bound Δ. Additionally, it can be assumed that the vertices of Q form a subset of the vertices of P. This is known as the *min–#* problem [123] and has been extensively studied [124]. According to [124], the most widely used heuristic approximation is the Douglas-Peucker algorithm [114], also published as Ramer's algorithm [113]. This algorithm consists of iteratively refining a coarse approximation Q until the maximum distance of any point on P from the segment on Q which represents it is less than the specified error tolerance. In addition to specifying the error tolerance, the starting point of Q has to be decided as the quality of the approximation varies depending on it. A number of heuristic approaches exist for choosing this point.

The above framing of the problem does not assume that the given polygon, P, has been corrupted due to digitisation noise and this is true for most polygonal curve approximation techniques [125]. Theoretically, this is problematic if one wishes to recover the true polygon that P represents because as the error tolerance is reduced the approximation approaches the noisy polygon. In practice, however, such approximations are used to reduce noise and improve the reliability of shape measures such as rectilinearity [122].

5.3.2 Discrete Curve Evolution

The approach used here, which takes a different view of the problem, is one of discrete curve evolution (DCE), as proposed by Latecki & Lakämper [126]. Discrete curve evolution works as follows: at each stage in the evolution process the least relevant or conspicuous vertex of the polygon is removed to produce a new, simplified polygon (with the two line segments which formed the vertex having been replaced by a single segment). The relevance of each vertex forming the new polygon is then calculated and the process repeated.

Vertex relevance, or conspicuousness, is defined by Equation 5.8 which refers to Figure 5.13. Essentially, the more relevant a vertex is the more it contributes to defining the shape of the polygon of which it is a part. Since Equation 5.8 depends on normalised lengths and the turn angle is relative to the line segments themselves,

the evolution is scaling, rotation and translation invariant.

$$K(s_1, s_2) = \frac{\beta(s_1, s_2) l(s_1) l(s_2)}{l(s_1) + l(s_2)} \tag{5.8}$$

where

$K(s_1, s_2)$ = relevance of the vertex created by line segments s_1 and s_2

$\beta(s_1, s_2)$ = the turn angle in radians from s_1 to s_2

$l(s_1), l(s_2)$ = lengths of line segments s_1 and s_2 normalised with respect to the perimeter length (the sum of the lengths of each line segment forming the polygon, including $l(s_1)$ and $l(s_2)$)

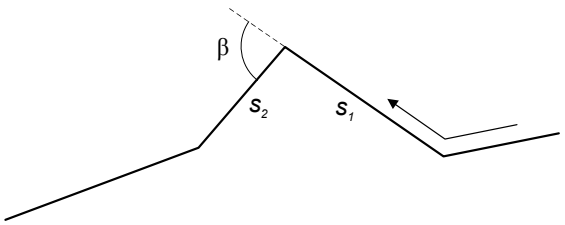


Figure 5.13: Line segments and turn angle used in calculating vertex relevance, adapted from [126].

As the evolution process proceeds the curve is gradually simplified and detail is removed. The result is a set of curves at multiple scales. In the early stages of the simplification digitisation noise is removed as it tends to take the form of vertices of little significance [126]. This results in the underlying shape being recovered. As simplification continues to increase more significant features of the curve are removed and the curve becomes an increasingly abstract representation of the underlying shape.

DCE has strong parallels with the anisotropic scale-space described earlier (and scale-spaces in general [127]) for extracting homogeneous regions, albeit applied to a curve instead of a 2D surface. In both cases a simplified form of the original signal is derived with increasing scale through the removal of fine detail. In the case of anisotropic diffusion a partial differential equation forms the basis for the evolution of the image; in the case of DCE, Equation 5.8 is the driver.

Latecki & Lakämper [126] state the following properties of DCE:

$P1$. Shape complexity is simplified.

$P2$. No shape rounding effects are introduced.

$P3$. Relevant features do not dislocate, that is, shift their position, over scale. This is because the vertices remaining after each elimination do not change position.

$P4$. The evolution is stable with respect to noise deformations and noise elimination takes place in the early stages.

$P5$. Straight line segments are extracted from noisy boundaries due to the repeated linearisation process that occurs with vertex removal.

All of the above properties are important for shack detection. Reducing shape complexity ($P1$) facilitates verification and grouping of hypotheses. It is undesirable to have the simplification process introduce artifacts such as shape rounding, especially considering that shack roofs are assumed to be rectilinear ($P2$). For accurate detection, it is important that the roof boundaries are not shifted at different scales ($P3$). The simplification process is directly used for the removal of digitisation noise ($P4$). Shack and building outlines are expected to be composed of straight line segments, so using a simplification method that will implicitly produce these is beneficial ($P5$).

Even though simplification results in linearisation of the shape boundary, the degree of modelling incorporated in the technique is relatively small, in the sense that the linearisation process does not rely on knowledge of a particular building model. This implies that various building shapes can be extracted. In model-driven simplification (Section 5.5), however, a specific model is assumed.

Note that properties $P1$ and $P2$ do not hold for other forms of curve evolution based on diffusion models. For example, Gaussian blurring, in which at each iteration the co-ordinate of every vertex is averaged with its immediate neighbours, does not obey $P2$ and $P3$.

5.3.3 Choosing a Stopping Point

A key issue in using vertex relevance for simplification is deciding on a stopping point for the evolution of the curve. For further detection steps to be successful a stopping point needs to be chosen so that digitisation noise is eliminated and perceptual appearance is sufficient for shack recognition.

Here, a simple, pragmatic approach is taken to determine the stopping point. This approach directly addresses the influence of digitisation noise. Although the vectorisation of the boundary produces line segments at specific orientations, the "true" boundary of a shack could have line segments which are not confined to these orientations, and indeed could have segments which are at none of these orientations. Figure 5.14a illustrates the original boundary derived from a homogeneous region from the Marconi Beam image, while Figure 5.14b presents a histogram (with a bin size of 1°) of the orientations of the boundary segments. The boundary has a clockwise direction so the segment angles fall in the range [0°, 360°). These angles are mapped to the first quadrant (the range of the histogram) by reflecting second quadrant angles through the y-axis, third quadrant angles through the 315° line, and fourth quadrant angles through the x-axis. As can be expected all of the line segments of the original boundary occur at 0°, 45° and 90°.

To reduce the effect of digitisation, DCE is used to simplify the boundary until the frequency of boundary segments which lie at 8-connected angles is reduced to values in line with other orientations. Initially, the maximum frequency of angles other than the 8-connected angles is 0 (Figure 5.14b). As DCE proceeds, all points with a relevance of 0, that is, redundant points that lie on straight lines, are eliminated and the shape remains unchanged. After all redundant points have been removed, the elimination of vertices brings about shape simplification and segments at new orientations appear. The stopping point is chosen to be when the maximum frequency of line segments at 8-connected angles is no greater than the maximum frequency of segments at other angles. The simplified boundary at this point is given in Figure 5.14c along with the histogram of its line segment orientations, Figure 5.14d. It is apparent that digitisation noise has been removed, yet the essential features of the shape remain. The histogram depicts that the boundary segments which lie at 8-connected angles no longer dominate.

5.3.4 Boundaries After Noise Removal

Figure 5.15 presents some additional boundaries selected from the Marconi Beam image. The left column illustrates the original boundary for an extracted region while the right column presents the boundary after removing digitisation noise. All boundaries are overlaid on the ground truth area for the shack in question.

Figure 5.15a presents a boundary corresponding to a four-sided shack. The localisation is mediocre. All four corners are present and there are no major deformations with

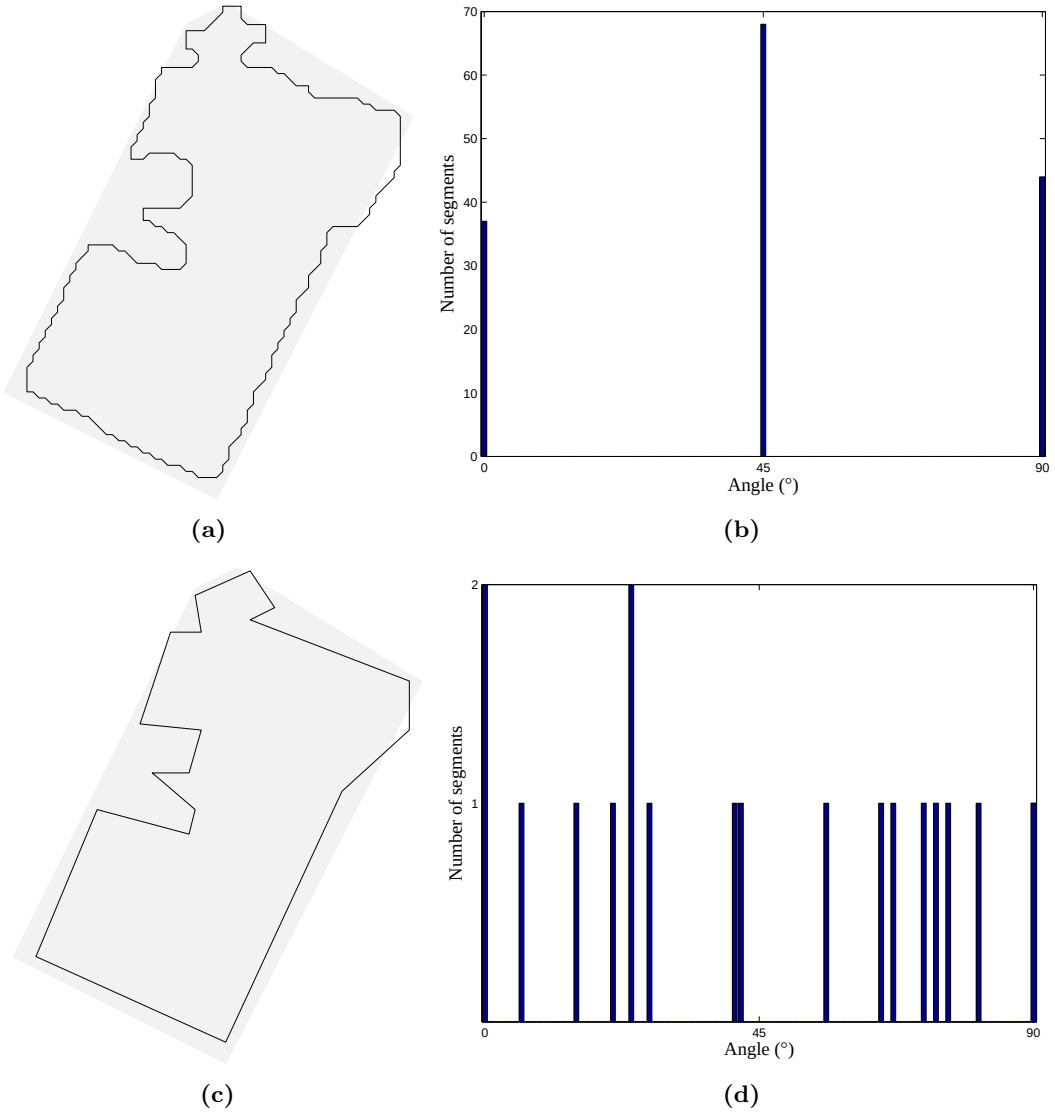


Figure 5.14: Boundaries and corresponding histograms of line segment orientations. (a) the original boundary – 149 vertices; (c) the simplified boundary after removing digitisation noise – 18 vertices; (b) and (d) histograms of the line segment orientations for the boundaries in (a) and (c).

respect to the ground truth. The boundary in Figure 5.15c corresponds to a 6-sided shack roof. It exhibits a sizable deformation due to reverse-flooding (background flooding into the shack roof). Figure 5.15e represents a rectangular shack with flooding at one of the corners which removes the corner point. Figure 5.15g displays a very poorly localised boundary having a large degree of flooding and reverse-flooding and Figure 5.15i is the same as given earlier in Figure 5.14a. In all cases, it can be observed that the heuristic approach for determining the stopping point gives simplified but not overly abstracted boundaries.

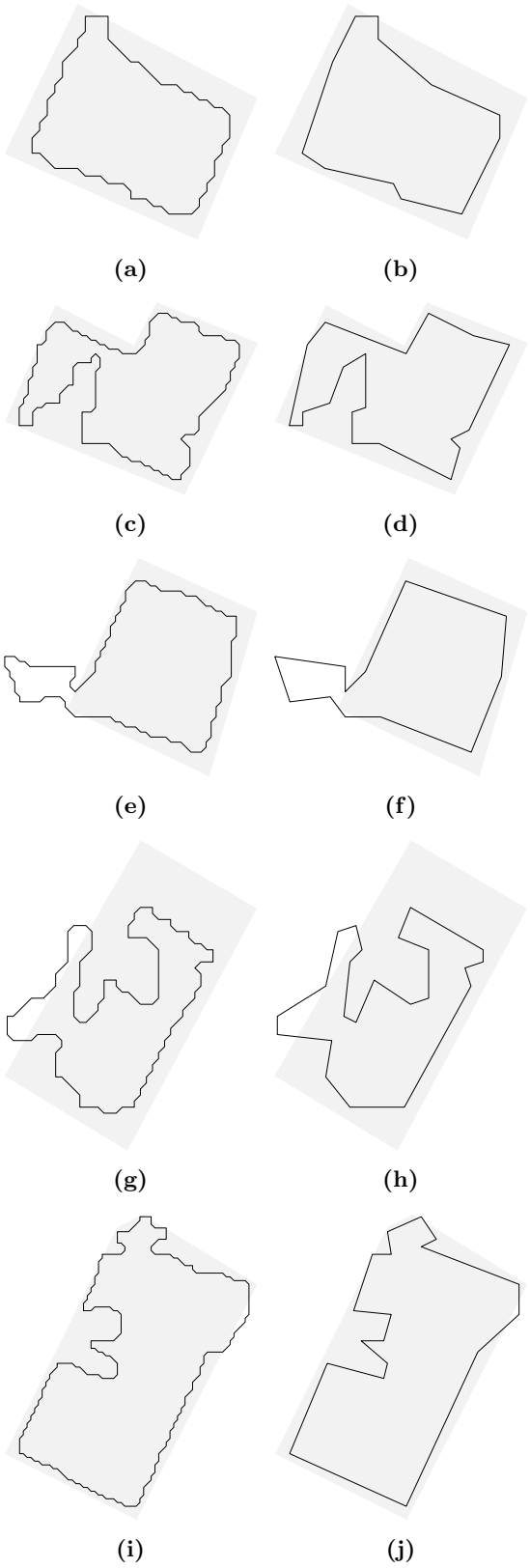


Figure 5.15: Selected boundaries from the Marconi Beam image exhibiting interesting features. LEFT COLUMN: Original boundary; RIGHT COLUMN: Boundary after removing digitisation noise. In each diagram, the shaded area represents the ground truth for the shack in question.

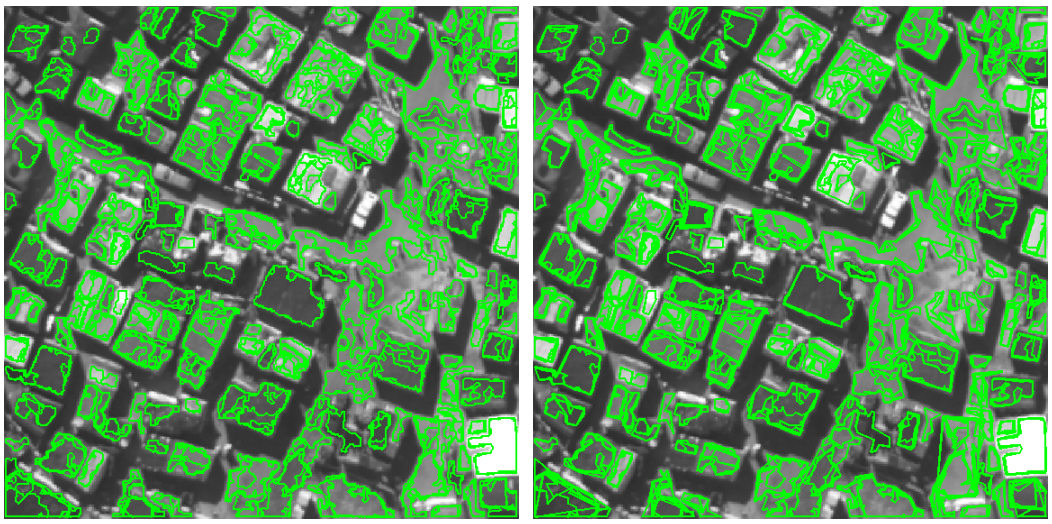
Most of the boundaries sit slightly within the interior of the shack roofs. This is an artifact of the homogeneous operator in that the homogeneous regions extracted do not extend right up to the shack roof borders. The reason for this is that the borders tend to correspond to image edges and thus do not satisfy the homogeneity constraint.

The boundaries in Figure 5.15 are not shown to scale. However, they vary quite substantially in length and in the area that they surround. Figure 5.15a is the shortest boundary enclosing the smallest area while Figure 5.15i is the longest, enclosing almost three times the area. The process for removing digitisation noise is not overly sensitive to scale and performs well in spite of these large differences. Note, the stopping point cannot be usefully determined by using a fixed threshold for vertex relevance and continually removing vertices until none lie below the threshold. In this case, the scale of the boundary does come into play. A threshold which is appropriate for a short boundary becomes inappropriate as the boundary increases in length because although the scale of the boundary increases, the scale of the perturbations due to digitisation noise do not. Vertices which are products of digitisation noise on a short boundary affect the shape of the boundary more substantially, and consequently, have higher relevance values than those on a much longer boundary. Therefore, a threshold chosen to stop the simplification of a short boundary after digitisation noise is removed will result in the oversimplification of a long boundary.

It is also evident that DCE linearises the boundary during simplification. This is one of its useful properties as we expect shack hypotheses to be composed of linear segments.

It can be argued that the boundaries still exhibit artifacts of vectorisation because simplification is stopped before all the segments at the 8-connected angles are removed. This is deemed to be acceptable as digitisation noise is substantially diminished, while highly relevant vertices, such as corners, are maintained. This allows the canonical orientation (Section 5.5.1) of each hypothesis to be reliably determined, which is required for model-driven simplification.

Figure 5.16 illustrates, on the left, the boundaries of the filtered, extracted regions (see Figure 5.8) from all stack levels. On the right, the noise-free versions of the boundaries are given.



(a) Extracted region boundaries in green.

(b) Region boundaries after digitisation noise removal.

Figure 5.16: Extracted region boundaries and noise-free boundaries for the Marconi Beam image. Boundaries from *all* stack levels are superimposed on the source image.

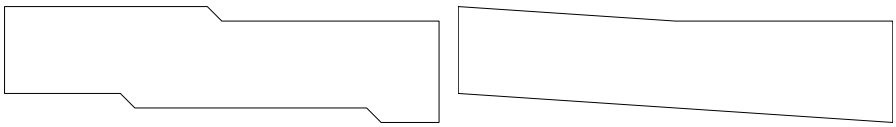
5.3.5 Handling Boundaries Aligned with 8-Connected Angles

In the explanation above it is assumed that the boundaries being dealt with are not genuinely aligned with the 8-connected angles. This assumption leads to the conclusion that polygonal segments with orientations matching these angles are unwanted products of vectorisation and need to be removed. Nevertheless, shacks or buildings and their corresponding region boundaries, may in fact, predominantly align with the 8-connected angles. It is important that the noise removal process is capable of handling these special cases.

It is necessary to discriminate between genuinely aligned boundaries and the rest in order to remove noise appropriately. If a boundary *is aligned* with the 8-connected angles then it is expected to consist of predominantly long line segments at orientations of *either* 0°and 90°*or* 45°(angles are projected into the first quadrant as for the histograms given earlier). This is not the case for a boundary at an unaligned orientation.

Discrimination takes place through the DCE process itself. Recall that the stopping point is determined to be when the maximum frequency of 8-connected line segment angles is no greater than the maximum frequency of line segments at other angles. However, if long line segments at the 8-connected angles dominate then few segments emerge at other angles during the DCE process. This causes the boundary to be

over-simplified, and its actual shape obscured, by the time the stopping point is reached. In these situations, it is observed that the stopping point is reached with only 3 vertices or less remaining — this is used to identify boundaries aligned with the 8-connected angles. To prevent over-simplification, if a boundary is simplified to 3 vertices then the stopping point threshold is doubled to allow the maximum frequency of 8-connected line segments to be double that of segments at other angles. This is illustrated in Figure 5.17 for a boundary composed of mostly horizontal and vertical line segments.



(a) Original region boundary largely composed of horizontal and vertical line segments.

(b) Boundary after removing digitisation noise.

Figure 5.17: Noise removal on a boundary aligned with the 8-connected angles.

5.3.6 DCE Compared to Other Noise Removal Techniques

As shown in Table 3.1 a number of different approaches have been used for digitisation noise removal. Here, the DCE approach is compared to these other approaches.

In [114, 113, 78] noise removal is framed as polygonal approximation to within a certain error tolerance. In DCE, noise is seen as fine scale detail distorting a shape's boundary. Simplification based on vertex relevance is used to uncover the underlying shape. DCE is classified as a *merge* method as vertices are repeatedly removed resulting in the merging of adjacent segments. The method in [78] is also a merging algorithm as vertices are removed based on their distance to an approximating line segment. The polygonal approximation methods of [114, 113] are termed *split* methods as initial approximation is repeatedly split into more line segments to reduce the approximation error.

Both merge and split approaches linearise the original boundary. Key issues for split methods are determining a suitable starting point and an error threshold. The key issue in the DCE approach is determining the stopping point. Here, an intuitive approach, based on the knowledge of the noise artifacts introduced by vectorisation, has been used.

DCE is related to the general idea of information abstraction through scale-spaces and complements the use of anisotropic diffusion in this system. DCE also provides a natural transition to the follow-on stage of model-driven simplification, as the model-driven simplification algorithm is itself a merging algorithm.

Finally, there is the issue of the scale of the boundary being dealt with. Here "scale" is being used to refer to the size of the region that the boundary encloses. The dominant corner detection method in [111, 107] relies on a Gaussian filter of fixed width to remove the effects of noise. This method is shown to be particularly vulnerable to changes in boundary scale. On the other hand, noise removal is fairly robust for large changes in scale when using DCE and a stopping point based on line segment orientation.

5.4 Hypothesis Verification Using Shadow

In the previous section, it is described how digitisation noise is removed from the extracted region boundaries. These boundaries are taken to be shack hypotheses. It is now necessary to verify that the hypotheses correspond to 3D objects. This is done through the use of shadow. Recall that the sun vector's direction and the typical width of a shack-cast shadow are known through manual inputs (Section 5.1.1).

Firstly, the shadow-casting sides of each boundary are determined. This is done by calculating each boundary segment's midpoint and moving a nominal distance along the direction of the sun vector, as in [45]. If the new point lies outside the boundary then there exists the potential for shadow to be found alongside the segment. Accordingly the segment is marked as part of the roof-shadow section of the boundary. The roof-shadow part of the boundary need not be continuous. Once the roof-shadow segments of a boundary are determined, equidistant points along these segments are generated and the sun vector is translated to each of these points. The image is then sampled (at sub-pixel resolution) at ten points along the sun vector at each of its translated positions. This is illustrated in Figure 5.18. Experimentally, it was determined that ten samples provides sufficient resolution for establishing the presence of shadow. The number of samples can be increased but the computational cost will increase as well.

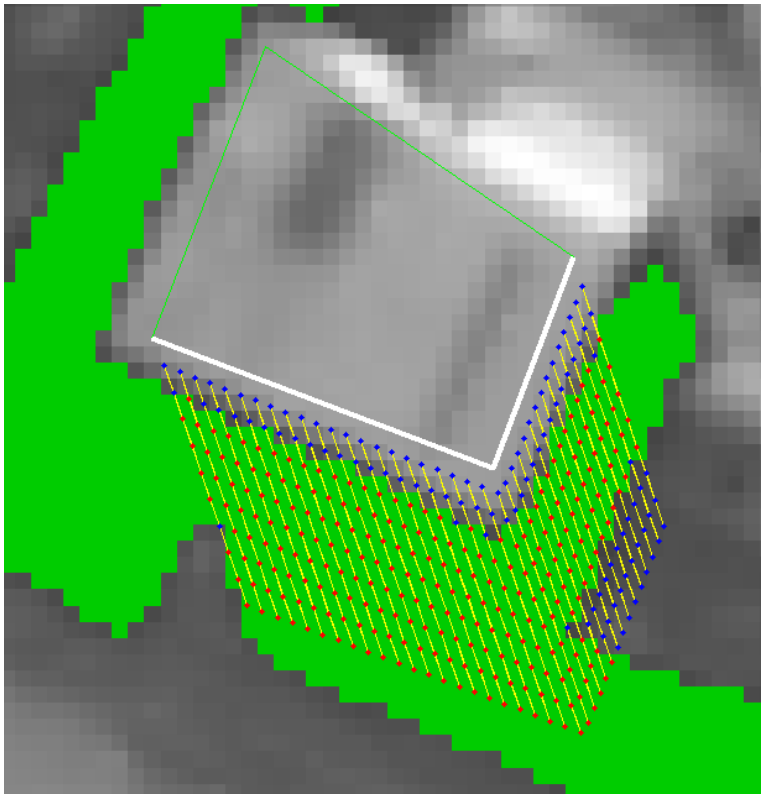


Figure 5.18: Using shadow for hypothesis verification. The roof-shadow section of the boundary is coloured in white while the non-roof-shadow section is in light green. Shadow pixels are coloured in dark green. The multiple positions of the sun vector alongside the shadow-casting line segments are shown as yellow lines. The sample points are shown as dots, a red dot indicates that the underlying pixel that is sampled is shadow while a blue dot indicates a sample where no shadow is found.

A support score is calculated for each hypothesis as follows:

$$\text{Support} = \left(\frac{S_{det} - S_{non}}{S_{tot}} + 1\right) * \frac{RS_{det}}{RS_{tot}} \tag{5.9}$$

$$S_{det} = S_{shadow}; \quad S_{non} = S_{no-shadow} \tag{5.10}$$

where

S_{det} = samples classified as detections

S_{non} = samples classified as non-detections

S_{tot} = total number of sample points

RS_{det} = number of roof-shadow line segments on which a detection occurs

RS_{tot} = total number of roof-shadow line segments

S_{shadow} = samples at which shadow is found

$S_{no-shadow}$ = samples at which no shadow is found

Equation 5.9 is the *general* equation used for calculating different kinds of support. It depends on counts of samples which have been classified as either "detections" or "non-detections". The higher the support score value, the greater the support that exists for the hypothesis in question. For the particular case of calculating *shadow* support, detections are the number of shadow samples and non-detections are the number of non-shadow samples (Equation 5.10). Later on in the detection process, the support equation is used to calculate other kinds of support.

Samples along each sun vector are analysed in a specific manner. The sample sequence is important with samples being processed in the direction of sun illumination. For a particular sun vector:

- All non-shadow samples are counted, as non-detections, from the start of the vector until either the first shadow sample is encountered or the end of the vector is reached.
- If a shadow sample is encountered, then this sample together with any *contiguous* shadow samples along the vector are counted as detections.
- If a sample further along the vector is encountered which is not shadow then it terminates the sample count for that particular sun vector.

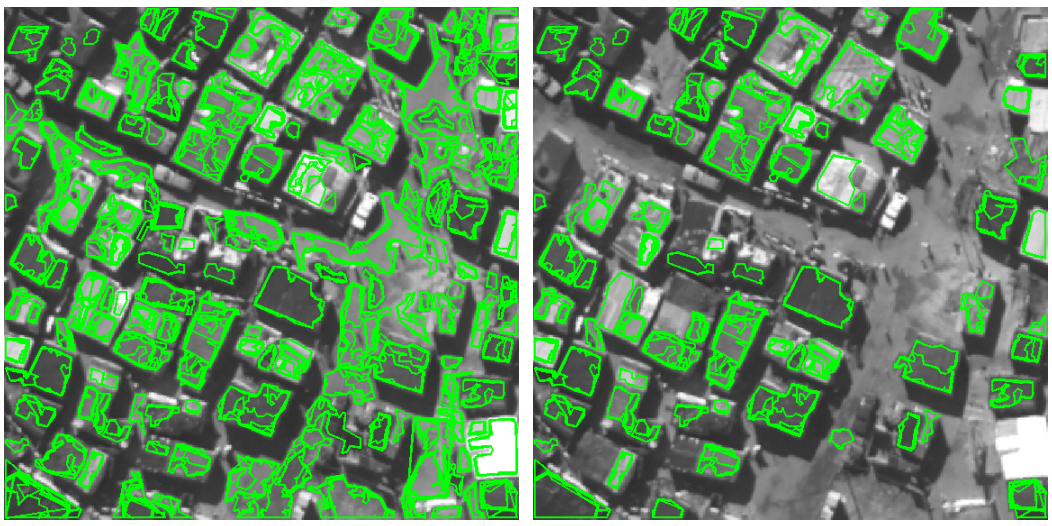
The reasoning behind this is as follows. As can been seen in Figure 5.18, the simplified boundaries usually sit slightly within the shack roof interiors. This means that the first samples encountered along the sun vector will be non-detections. These count as negative evidence for the hypothesis and the further displaced the roof-shadow line segments are from the start of the shadow area, the greater the amount of negative evidence that will be incurred. As one moves further along the sun vector and encounters a shadow area, shadow detections take place and these are counted as positive evidence for the hypothesis. In some cases the length of the sun vector will extend beyond the shadow cast by the shack. The non-detections which occur as a result of this are not counted as this would unfairly penalise shacks whose shadows are not fully visible due to occlusions from neighbouring shacks. This situation can be observed in the bottom-right part of Figure 5.18. The non-detections (blue dots) in this area are not counted when determining the support score.

The term $\frac{S_{det} - S_{non}}{S_{tot}}$ produces values in the range $[-1, 1]$. Negative values are produced if the number of non-detections exceeds the number of detections, or the negative evidence exceeds the positive evidence. In order to ensure that the support value is always positive, one is added to this term. The multiplication factor in Equation 5.9 is a penalty factor. It is expected that detections (in this case shadow)

occur along each and every roof-shadow line segment. If this is indeed the case then RS_{det} will equal RS_{tot} and the term will evaluate to one and not lower the support score. However, if a roof-shadow line segment is present which does not have any associated detections then the support score is penalised. The penalty factor dominates the equation when the number of roof-shadow line segments is small. For example, if there are 4 roof-shadow line segments and shadow is found along only 3 of them, then the support score will be decreased by a 1/4.

When comparing boundaries against one another the support score tends to favour simpler boundaries with fewer sides which are well localised with respect to shadow. However, the score does not account for the length of the roof-shadow line segments. Assume that there are two line segments, one which is far longer than the other, and that they both enjoy full shadow support with $S_{det} = S_{tot}$. The support values will be identical even though the longer line is more noteworthy and probably a better indicator of a valid building hypothesis. This is taken into account, later on, in the hypothesis selection stage (Section 5.6) where size is a factor.

A shadow-support threshold is used to discriminate between hypotheses with reasonable support and hypotheses without. This is illustrated in Figure 5.19. On the left, all of the simplified hypotheses' boundaries are given; on the right, only the boundaries that have adequate support are displayed.



(a) Region boundaries after digitisation noise removal.

(b) Boundaries remaining after shadow verification.

Figure 5.19: Noise-free boundaries and shadow-verified boundaries for the Marconi Beam image. Boundaries from all stack levels are superimposed on the source image.

In this system, shadow is used only for verification and not for height estimation. This less rigourous use of shadow allows for shadow occlusions to be dealt with in an interesting manner. As mentioned earlier, the presence of shadow is exaggerated through dilation in order to widen shadows which are largely occluded by closely clustered shacks. This increases the support score for hypotheses with occluded shadows as more sample points fall on the dilated shadow regions. In systems where the length of the cast shadow is used for height estimation it is important to precisely delineate the shadow boundary. These restrictions do not apply here.

5.5 Model-Driven Simplification

The shadow-verified boundaries — candidate hypotheses — surround homogeneous image regions which correspond, with varying degrees of fidelity, to actual shack roofs. In addition, multiple hypotheses overlap as hypotheses are generated from each image in the image stack. In order to remove large distortions of the boundaries, and facilitate further verification and grouping, model-driven simplification of the boundaries is performed.

Minor boundary distortions due to digitisation noise have been removed in the previous stage. This allows shape measures to be reliably calculated. The two measures used, compactness and rectilinearity, as well as the model-driven simplification algorithm which depends on them, are described below.

5.5.1 Rectilinearity

Žunić & Rosin [86] have proposed the following rectilinearity measure for closed polygons[2]:

$$\mathscr{R}(P) = \frac{4}{4-\pi} \cdot \left(\max_{\theta \in [0,2\pi]} \frac{\mathscr{P}_e(P)}{\mathscr{P}_{cb}(P,\theta)} - \frac{\pi}{4} \right) \tag{5.11}$$

where

$\mathscr{P}_e(P)$ = Euclidean perimeter of polygon P

$\mathscr{P}_{cb}(P,\theta)$ = city-block perimeter of P rotated by angle θ with the origin as the centre of rotation

[2] They have proposed 4 measures in total, two in [86] and an additional two in [122]. All measures are shown to have similar performance. The measure given in the main text is the first measure that they proposed and the one used in this system.

This measure produces values in the range $(0, 1]$ and is invariant under rotation, translation and scaling transformations. It is further explained in the following paragraphs.

Given two points (x_1, y_1) and (x_2, y_2), the Euclidean length is

$$l_e = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

while the city block length is $l_{cb} = |x_1 - x_2| + |y_1 - y_2|$. Determining the lengths in this manner for each segment of a polygon and adding them gives the respective perimeters, $\mathscr{P}_e(P)$ and $\mathscr{P}_{cb}(P)$. The Euclidean perimeter is constant and independent of rotation whilst the city-block perimeter varies as the polygon's orientation changes with respect to the x and y axes. For a polygon which is perfectly rectilinear, that is, all of its inner angles belong to the set $\{90°, 270°\}$, the city-block perimeter will be at a minimum (and identical to the Euclidean perimeter) when the polygon is rotated so that its sides are vertical and horizontal and align with the x and y axes. In the context of Equation 5.11 the maximum value of rectilinearity will be achieved, which is one.

Moving to polygons which are not perfectly rectilinear, it has been proven that the maximum value of the term $\max_{\theta \in [0,2\pi]} \frac{\mathscr{P}_e(P)}{\mathscr{P}_{cb}(P,\theta)}$ will occur when the rotation angle θ is such that at least one side of the polygon is aligned with either the x or y axes [86]. This is an important result for two reasons. Firstly, it greatly simplifies the calculation of rectilinearity as θ does not have to vary continuously from 0° to 360° — it can be restricted to the set of angles that align each side of the polygon to the nearest x or y axis. Secondly, the angle θ which corresponds to the rectilinearity value is useful in characterising the orientation of the shape in question. This *canonical* orientation appears to be a better descriptor of orientation for fairly rectilinear shapes than one based on second order moments [122]. This is because the rectilinearity measure is more attuned to local shape properties than moment calculations which are derived from gross spatial distribution.

5.5.2 Compactness

Compactness is a widely used, global shape measure and is calculated as follows:

$$\mathscr{C}(P) = \frac{4\pi \mathscr{A}(P)}{(\mathscr{P}_e(P))^2} \tag{5.12}$$

where

$\mathscr{A}(P)$ = area of polygon P

$\mathscr{P}_e(P)$ = Euclidean perimeter of P

The maximum value of compactness occurs for a circle. Compactness decreases as a shape becomes more irregular and narrow. In the formulation above, the measure produces a maximum value of one and has the range $(0, 1]$.

5.5.3 Algorithm

It is assumed that shacks are compact 4, 5 or 6 sided structures exhibiting a high degree of rectilinearity. This is a fair assumption based on the characteristics of informal settlements [6]. The algorithm is constructed in such a way so as to force the input boundary to fit this model solely through vertex removal. It is presented below as Algorithm 5.1.

Input: SB – boundary after removing digitisation noise {see left column in Figure 5.21}
Output: MB – boundary after model-driven simplification {see right column in Figure 5.21}

$i \leftarrow$ total number of vertices of SB
if $i \leq 4$ **then**
  $MB \leftarrow SB$ {no further simplification possible}
else
  while $i > 4$ **do** {simplify to quadrilateral}
    for $j = 1$ to i **do**
      delete vertex j of SB to create new boundary SB_j
      if $\text{abs}(\theta_{SB_j} - \theta_{SB_{in}}) \leq R$ **then** {determine difference in canonical orientations}
        $O_j \leftarrow \mathscr{R}(SB_j) + \mathscr{C}(SB_j)$ {O calculated using Equations 5.11 and 5.12}
      else
        $O_j \leftarrow 0$ {rotation threshold exceeded}
      end if
    end for
    $SB \leftarrow SB_j$ where j gives $\max_{j \in [1,i]} O_j$ {replace SB}
    $i \leftarrow i - 1$ {SB has one less vertex}
  end while
  $MB \leftarrow SB$
end if

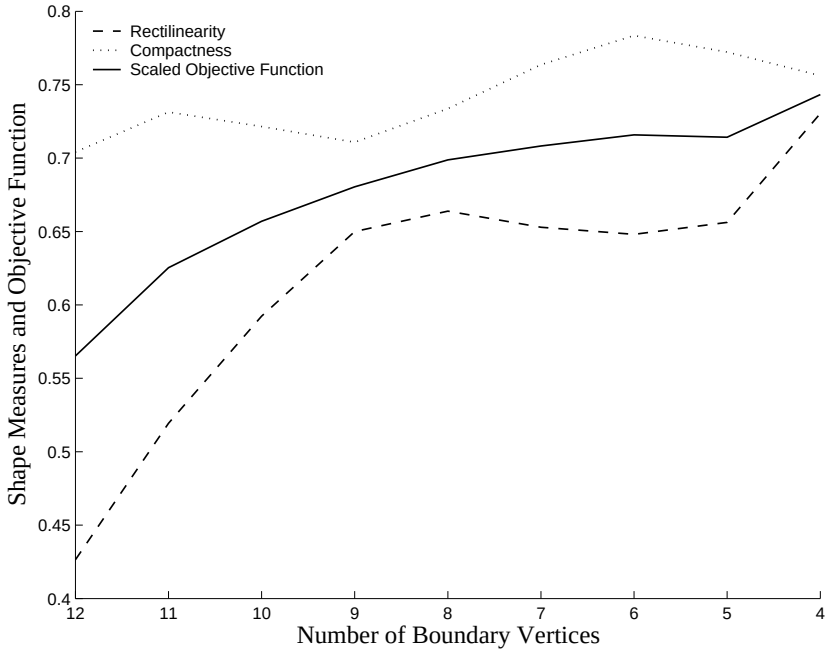
Algorithm 5.1: Algorithm for model-driven simplification. Comments are included in braces.

The following points are worth noting:

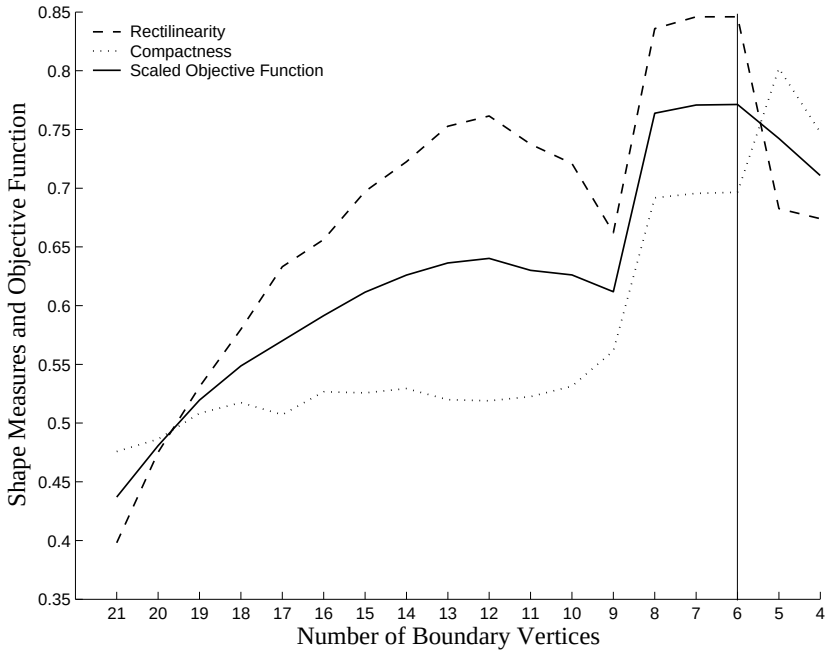
- The input is a boundary from which digitisation noise has been removed while the output is a simplified version of this boundary which conforms as much as possible to the assumed model.

- If the boundary only consists of 4 vertices it cannot be further simplified according to the model (lines 2–3).
- The simplification is iterative with each iteration (lines 5–16) producing a boundary with one less vertex. This boundary forms the input for the next iteration. Simplification terminates when the input boundary is simplified to a quadrilateral.
- In each iteration, every single vertex of the boundary is removed in turn (lines 6–13) and the objective function, O, associated with the resulting boundary is calculated. O is the sum of the rectilinearity and compactness measures. The "look-ahead" boundary with the maximum objective value is chosen as the new boundary for the next iteration (line 14).
- $\theta_{SB_{in}}$ is the canonical orientation (the orientation corresponding to the rectilinearity value) of the input boundary. If a particular vertex removal results in a new canonical orientation which exceeds the rotation threshold, R, then the objective function of the associated boundary is set to zero, removing it as a possible candidate for the next iteration (line 11). A maximum rotation of 15° is allowed. This, heuristically chosen, constraint ensures fidelity between the orientation of the simplified boundary and the input.

The canonical orientation of the input boundary is parallel to one of its line segments. This means, in effect, that neither of the two vertices forming this segment are likely to be eliminated as the resulting boundary's canonical orientation would exceed the rotation threshold (unless there is another significant line segment at a similar orientation). Boundaries are simplified until only four vertices remain. A direct consequence of these facts is that it is always possible, in every iteration, to find a vertex to remove which does not change the canonical orientation and results in a non-zero value for the objective function. This does not mean, necessarily, that the function value will increase from one iteration to the next. For 4-sided shacks in which the noise-free boundary is well-localised it is possible to expect the objective function to generally increase as the boundary is regularised, as shown in Figure 5.20a. For boundaries which are not well-localised to begin with, and for boundaries corresponding to 6-sided shacks in particular, this will not be the case. For example, the objective function will generally peak for 6-sided shacks (Figure 5.20b) when the boundary is reduced to six sides. Further simplification will decrease the rectilinearity and hence the objective function value. Therefore Algorithm 5.1 is extended to cater for 6-sided shacks by storing the simplified boundary and its associated objective function values at 4, 5 and 6 vertices. From these, the boundary with the highest value of O is selected as the final boundary.



(a) Variation in shape measures for a boundary corresponding to a 4-sided shack, in this case, the boundary in Figure 5.15a is used.



(b) Variation in shape measures for a boundary corresponding to a 6-sided shack, in this case, the boundary in Figure 5.15c is used. The objective function peaks with 6 vertices remaining

Figure 5.20: Rectilinearity, compactness and the objective function as model-driven simplification occurs. The above graphs show how these values vary as the noise-free boundaries are transformed to model-driven simplified boundaries through vertex removal. The objective function has been scaled to a similar range as the other shape measures, in order to better illustrate on a single plot all three variables.

The above algorithm embodies a sub-optimal, heuristic solution to maximising both the rectilinearity and the compactness of the input boundary. It is capable of achieving locally optimal results, but not globally optimal results, as only the immediate neighbourhood of solutions is considered in each iteration. The algorithm is a "greedy" one in that the neighbouring solution that offers the maximum value in terms of the objective function is chosen at each iteration.

5.5.4 Boundaries After Model-Driven Simplification

In the left-hand column of Figure 5.21 selected noise-free boundaries derived from the Marconi Beam image are shown. The boundaries after model-driven simplification are given in the righthand column. In all cases relatively large distortions of the boundary are corrected. Additionally, the algorithm is able to approximately recover missing corner points provided that a boundary point exists near the missing corner (Figure 5.21f). However, it is important to note that if the canonical orientation of the input boundary is not closely aligned with one of the shack's walls then the algorithm will produce a poorly localised boundary. This is evident in Figure 5.22 which displays the model-simplified boundaries overlaid on the source image. A few of the boundaries have longitudinal axes which deviate quite substantially from the longitudinal axis of the shack to which they correspond. This is most notable for a large rectangular boundary occurring on the left-hand side of the image just above centre.

The objective function weights the rectilinearity and compactness measures equally. It has been determined experimentally that using rectilinearity alone can result in degenerate boundaries which have near right angles but close to zero area. Additionally, maximising rectilinearity only can result in small stubs appearing where the boundary leaks into the background as shown in Figure 5.23a. Using compactness alone, forces boundaries to become more circular, which does not fit with the assumed model. Figure 5.23b illustrates this case.

One of the limitations of the model-driven simplification is that modelling occurs only through the removal of vertices. This has the consequence that a starting boundary which does not adequately cover a shack roof cannot be expanded (see Figure 5.21h). In order to attain better roof coverage, boundaries are expanded using the technique given in Section 5.8.

In much of the literature on regularisation approaches (see Section 3.5), the distortions

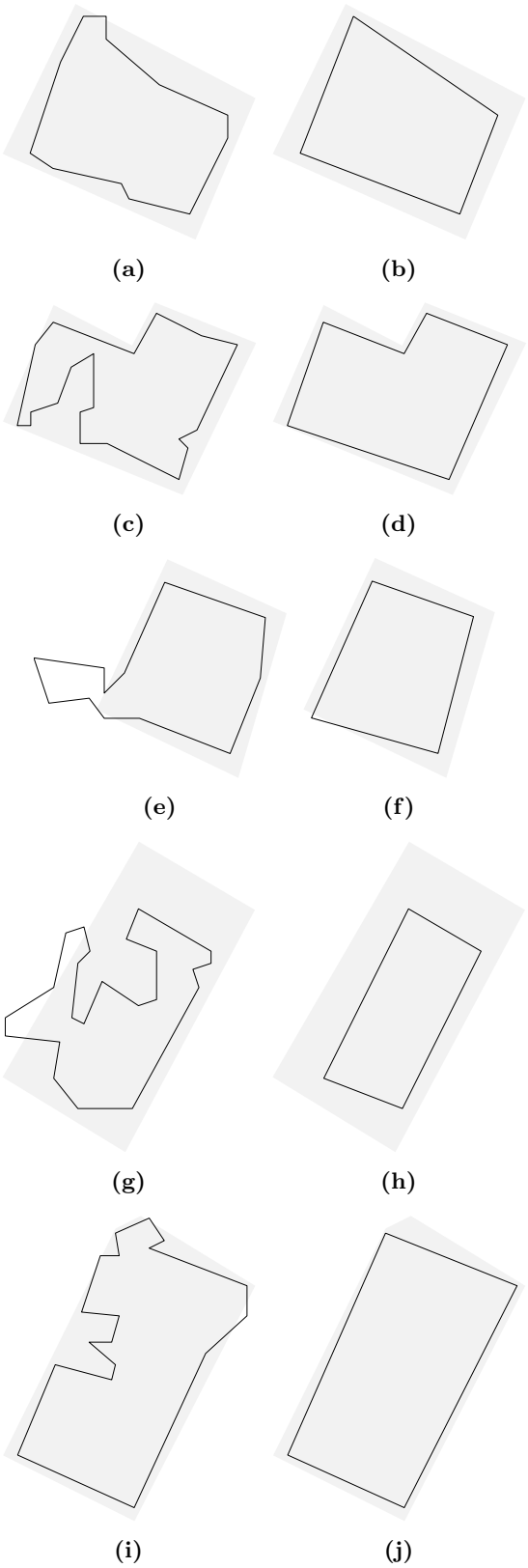
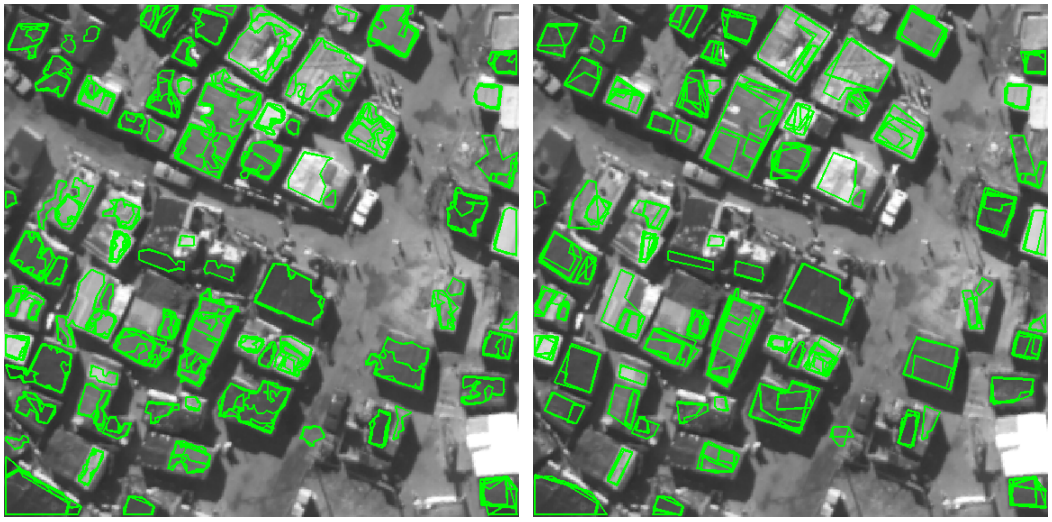


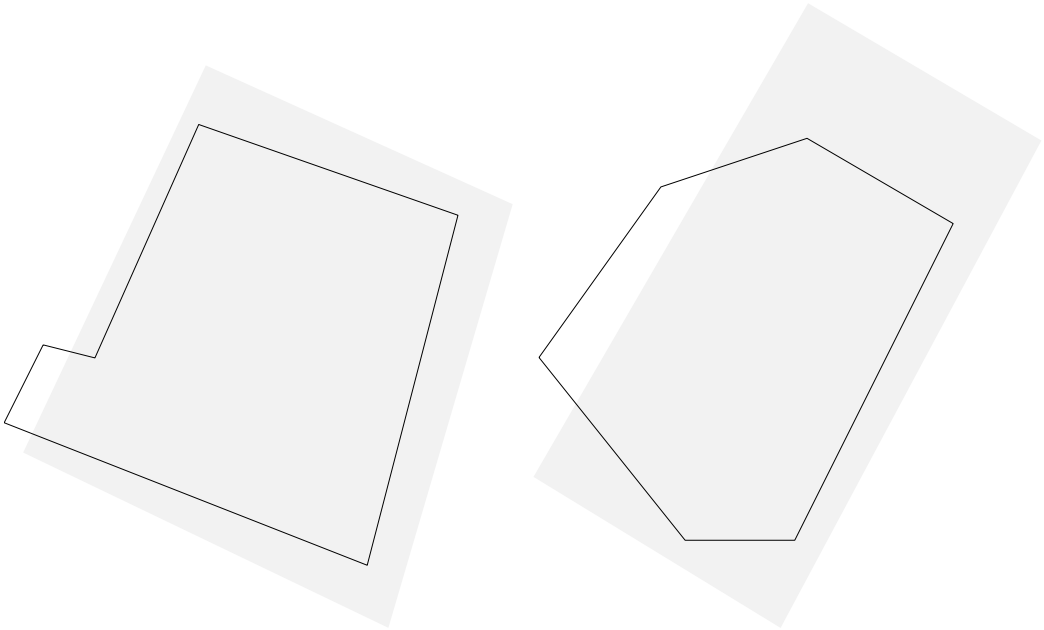
Figure 5.21: Model-driven simplification of selected boundaries from the Marconi Beam image. LEFT COLUMN: Boundaries after removing digitisation noise; RIGHT COLUMN: Boundaries after model-driven simplification. In each diagram, the shaded area represents the ground truth for the shack in question.



(a) Noise-free boundaries before model-driven simplification.

(b) Boundaries after model-driven simplification.

Figure 5.22: Shadow-verified boundaries for the Marconi Beam image before and after model-driven simplification. Boundaries from all stack levels are superimposed on the source image.



(a) Boundary produced by an objective function based solely on rectilinearity (from the noise-free boundary in Figure 5.21e).

(b) Boundary produced by an objective function based solely on compactness (from the noise-free boundary in Figure 5.21g).

Figure 5.23: Degenerate boundaries produced by maximising rectilinearity or compactness alone. In each diagram, the shaded area represents the ground truth for the shack in question.

that need to be corrected appear to be qualitatively less than those that are dealt with in this work. This need not mean that the documented regularisation methods cannot successfully correct such distortions. It is more a reflection of the fact that for many systems the starting boundaries are of higher quality, which is due to various factors including the data sources being used and the scenes being interpreted.

5.6 Hypothesis Selection Over Scale

As can be seen in Figure 5.22b, multiple boundaries exist corresponding to the same shack instance. Multiple boundaries arise because the shack roof is detected at different scales. This is illustrated in more detail in Figure 5.24 which depicts how a particular shack's boundary or, equivalently, the shack hypothesis, changes over scale. All of the hypotheses illustrated lie on a path from leaf node to root in the scale-tree (see Figure 5.12). In Figure 5.24, from left to right, top to bottom, the stack level at which the boundary has been extracted increases from lowest to highest. Interestingly, the hypothesis does not necessarily increase in size as blurring occurs. This is due to the fact that these boundaries have undergone model-driven simplification; the original region boundaries (which are not illustrated) do increase in size over scale. Importantly, all the hypotheses overlap each other spatially, offering different interpretations of the shack's boundaries. The following sections describe how these overlapping hypotheses are viewed in relation to one another and the mechanism that is used to select the single, best hypothesis over scale.

5.6.1 Regarding Overlapping Hypotheses as Mutually Exclusive

The goal of detection is to produce as accurate a shack boundary as possible. One approach, which is the approach adopted, is to regard the hypotheses as conflicting and mutually exclusive. An attempt is then made to identify the best of the overlapping hypotheses. This is described in more detail further on.

There are, however, other approaches to this problem if the overlapping hypotheses are not regarded as mutually exclusive. For example, the union of all the hypotheses or different aggregations of their common area may be considered [46]. This is feasible when the overlapping hypotheses are generated by different building extraction methods and are considered to be of equal standing, that is, they are all assumed to be equally valid. Here, this approach is less attractive because shacks are optimally detected at a single scale. This implies that the confidence in the validity of the

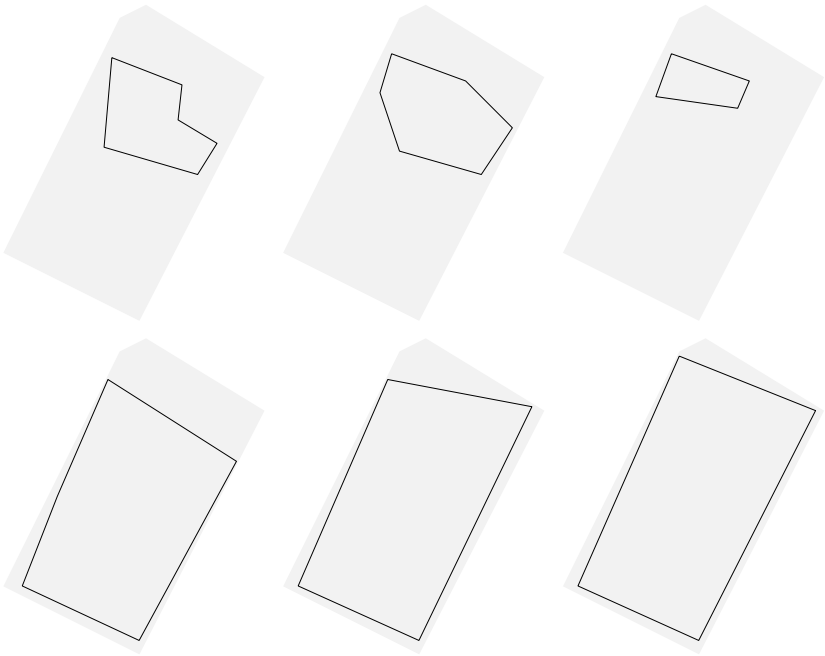


Figure 5.24: Evolution of a shack hypothesis over scale. The model-driven simplified boundary for a particular shack is shown at selected scales. From left to right, top to bottom, the scale at which the boundary has been extracted runs from fine to coarse. In each diagram, the shaded area represents the ground truth for the shack.

hypothesis for a shack varies depending on the scale. Taking the union of hypotheses across all scales does not take this into account and it is difficult to determine without a more detailed assessment which hypotheses should contribute to a common area.

An example of a sophisticated technique for resolving overlapping parallelograms in the image plane is given in [48, 128]. Overlapping hypotheses are classified as being duplicated, evidentially contained (one hypothesis is contained within another hypothesis and its supporting evidence is completely covered by the containing hypothesis), spatially contained but not evidentially contained, and overlapping but not contained. Some of the classifications result in the overlapping hypotheses being viewed as mutually exclusive, while in others not. This sophisticated analysis of overlapping hypotheses is possible because of the availability of fairly precise image evidence for each hypothesis, such as edge support, parallel edge support, corner support, and matched shadow corners.

In this system, given the manner in which hypotheses are generated, and linked over scale, most of the hypotheses on a path in the scale-tree tend to be evidentially contained (verified by the same shadow) by one of the larger hypotheses on the path. All overlapping hypotheses are regarded as competing and a fuzzy system is used to

evaluate both their supporting image evidence and their conformance with the shack model. In [48, 128] evidentially contained hypotheses are eliminated based solely on the fact that they share image evidence with the containing hypothesis; there is no need to evaluate model conformance as all hypotheses are parallelograms.

5.6.2 A Fuzzy Approach to Conflict Resolution

For each polygonal hypothesis, a number of derived attributes are produced:

- size (polygonal area),
- rectilinearity,
- compactness,
- shadow-support score (re-calculated for the model-driven simplified boundary).

It is possible to derive region statistics for the pixels contained within a hypothesis but these statistics are not expected to aid discrimination because, while some shack roofs appear fairly homogeneous, a significant number do not as they are composed of different roof materials.

These attributes can be seen as providing evidence for the hypothesis. Interpreting the evidence, in this context, may be defined as a measure of how well expectations regarding the hypothesis are met by actual observations. This evidence, on the one hand, may be positive in that it corroborates the fact that the hypothesis accurately delineates a shack. On the other hand, it may be negative in that it detracts from this fact. A scheme needs to be chosen which is capable of combining and reasoning with different kinds of evidence, all of which are represented numerically. The output of such a scheme will be a reflection of the system's confidence in the hypothesis. This will allow overlapping hypotheses to be ranked and the hypothesis with the highest likelihood to be selected as the final interpretation.

There are many different ways in which evidence may be combined. Some of the more well-known approaches include linear combination, neural networks, fuzzy logic, Bayesian methods (probabilistic approaches), and Dempster-Shafer reasoning [71, 24].

The most straightforward approach, linear combination, occurs when a linear weighted sum of the evidence components is calculated. Each component has a numerical value within a defined range. Each value is multiplied by a specific weighting and

the results are summed to produce an overall score. This score can be seen directly as a confidence value and thresholded to produce a crisp classification (i.e. shack or not-shack). Linearly combining evidence using weightings allows one to specify the relative importance of each component. Components which are regarded as more important will have higher weightings (see [48], for example). This type of approach is suitable when the score being calculated is based on the straightforward accumulation of image evidence (both positive and negative) for the hypothesis in question.

In the detection system at hand, however, other forces come into play. In Figure 5.22 it can be observed that many of the overlapping hypotheses within a set of overlapping hypotheses sit within the shack's borders. Strong image evidence for these smaller hypotheses often exists through high shadow-support scores. Larger hypotheses in the set which may, in fact, better delineate the shack may have less shadow support due to shadow occlusion. This is quite common because, as the length of the roof-shadow boundary increases, so does the likelihood of shadow occlusion. Therefore, the shadow-support score and hence the image evidence, on its own, is not a good indicator of the best-fitting hypothesis because hypotheses which are smaller in size are favoured. One way of dealing with this is to build into the score itself some compensation for size. Doing this, however, then obscures the meaning of the score which determines *support* but not *significance*.

The approach taken here, is to reason at a meta-level, about the significance of the support score. Using a linear combination of evidence was found to be inadequate in this scenario because of its limited flexibility and the fact that the weightings are fixed. A fuzzy logic approach is adopted in order to overcome this. It offers a great deal of flexibility, both with regard to the fuzzification of the input evidence through membership functions (which may be seen as non-linearly weighting each evidential component) and to the way in which evidence is combined through fuzzy "if-then" rules.

Fuzzy logic has been applied in many areas of image processing and computer vision including image segmentation, feature extraction, knowledge representation and processing, data fusion and classification [129]. Work by Mees et al. [130, 131] and Kang & Walker [132] is relevant here as they describe the use of fuzzy logic for reasoning about evidence pertaining to individual building hypotheses. In [131] fuzzy rules are used to evaluate hypotheses based on local image evidence such as edge length, strength and geometry, as well as region variance. Fuzzy rules determine the degree of membership of each hypothesis in the fuzzy set "house" based on

corroborating image evidence. Additionally, global fuzzy rules are used to evaluate inter-relationships between different houses, and houses and other objects like roads.

In [132] fuzzy logic is applied to the perceptual grouping of line segments. Geometrical relationships, such as, the angle between line segments and the smallest gap between segments' endpoints, are represented by fuzzy sets with handcrafted membership functions. The membership value of each set for a collection of line segments indicates how well the collection satisfies the constraint, termed the goodness value. Membership functions for different grouping configurations, such as collinear grouping, L-junction formation and so on, are generated by intersecting the membership functions for the appropriate geometrical relationships. A global goodness value, can, in turn, be computed from the goodness values of the constituent grouping configurations. This approach is used to group line segments into U-shaped structures which form building seeds.

5.6.3 The Fuzzy Inference System

The fuzzy inference system (FIS) used here involves the following steps [133]:

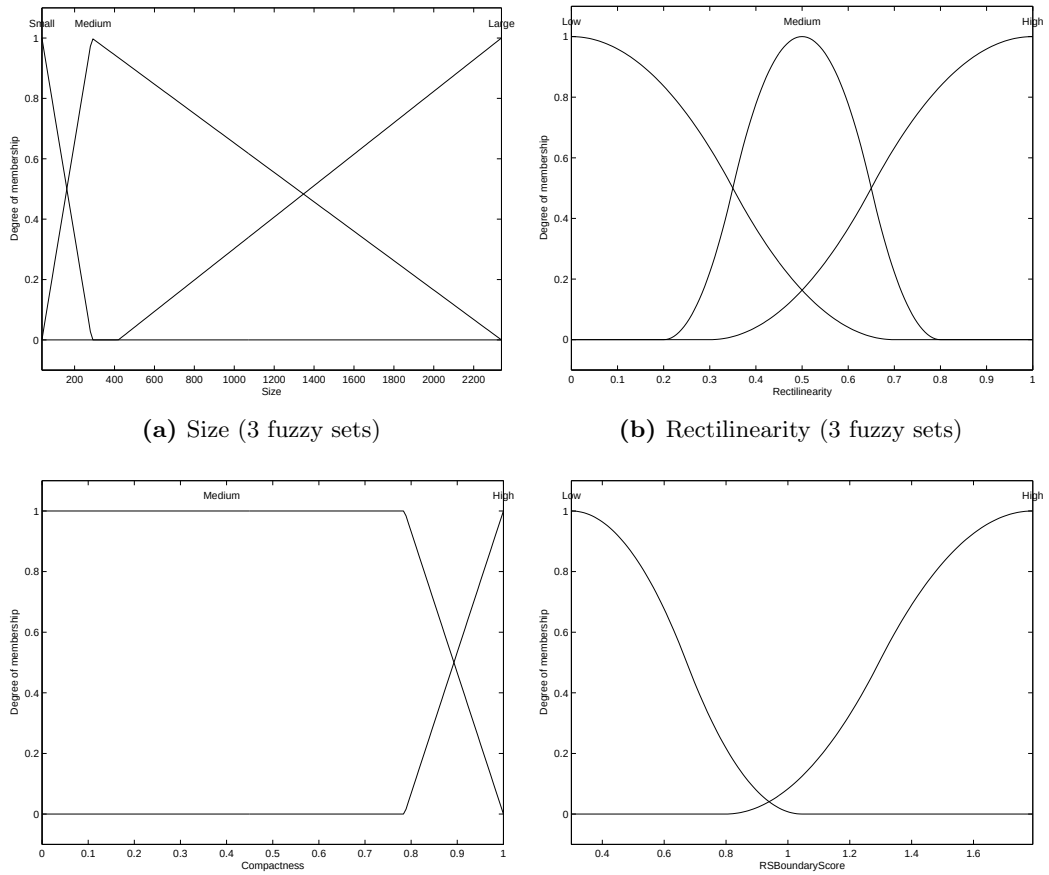
1. Fuzzifying the input variables.
2. Combining the fuzzified inputs in each fuzzy rule's antecedent clause through the use of fuzzy operators.
3. Implication from rule antecedent to the rule consequent.
4. Aggregation of the consequents from all rules.
5. Defuzzification of the output variable.

The next section describes the input and output variables in more detail, how their associated fuzzy sets are constructed, and fuzzification/defuzzification. Following this, the fuzzy rule base is presented along with the fuzzy operator, implication and aggregation methods used. A motivation for the choice of rules is also presented. The result of using the FIS to evaluate competing hypotheses is given in Section 5.6.4.

Input and Output Variables

A numeric variable, such as each of the attributes derived for a hypothesis, is described in terms of a fuzzy space. This space is composed of overlapping fuzzy sets which are defined by membership functions. Each of the membership functions corresponds

to a linguistic value for the variable. The total extent of the input space, from the smallest to the largest allowable value, is referred to as the *universe of discourse*. Figures 5.25 and 5.26 illustrate the different fuzzy input and output variables. As shown in Figure 5.25a, for example, there are three fuzzy sets associated with the *Size* variable and these correspond to linguistic values of "Small", "Medium" and "Large". The universe of discourse for *Size* ranges from the size of the smallest hypothesis to that of the largest.



(a) Size (3 fuzzy sets)

(b) Rectilinearity (3 fuzzy sets)

(c) Compactness (2 fuzzy sets)

(d) Support (2 fuzzy sets)

Figure 5.25: Membership functions for the FIS input variables. Size refers to the polygonal image area of a hypothesis; rectilinearity, compactness and support are calculated by the equations presented earlier. The universe of discourse for the size variable is [smallest polygonal area, largest polygonal area], for rectilinearity and compactness $[0, 1]$, and for the support score $[0, 2]$. In all graphs the y-axis indicates the degree of membership.

Each of the membership functions maps a given, crisp input to a membership value representing the degree of compatibility between the input and a particular linguistic value. In other words, membership functions determine the degree to

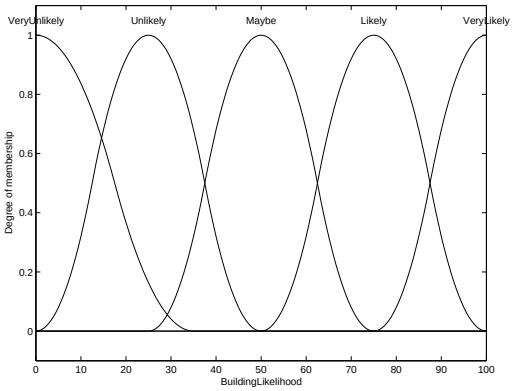


Figure 5.26: Membership functions (5 fuzzy sets) for the FIS output variable – Likelihood. The y-axis indicates the degree of membership. The universe of discourse is $[0, 100]$.

which the input is a member of the relevant fuzzy set. These functions range from zero to one with one representing 100% compatibility/membership. For example, a calculated rectilinearity of 0.5 would be entirely compatible with the concept "Medium Rectilinearity" whilst having a small compatibility with the concepts of "Low" and "High Rectilinearity" (Figure 5.25b). Note that fuzzy logic theory does not require the overlapping membership functions to sum to one at any point. Fuzzy sets are context-dependent and need to be determined specifically for an application. The next few paragraphs explain how the fuzzy sets have been chosen.

As alluded to earlier, hypothesis size is a critical attribute, partly because the shadow-support score favours small sizes. Empirically, it was found that three size categories (small, medium and large) were sufficient for making reasonable "size-based" decisions. The fuzzy sets corresponding to these categories are determined dynamically (on a per-image basis) by generating some basic statistics related to the sizes of all the conflicting hypotheses. From a histogram of these sizes, given in Figure 5.27 for the Marconi Beam image and representative of the images analysed, it is evident that there is a very high occurrence of smallish-sized hypotheses, although there are smaller hypotheses and a number of large-sized hypotheses (sometimes representing true positives in the form of formal buildings and large shacks). The "Medium" fuzzy set is shaped to identify the smallish-sized hypotheses. It is a triangular function, starting at the smallest size value, peaking at the median value, and terminating at the largest value (refer to Figure 5.25a). "Small" and "Large" identify hypotheses which are respectively smaller and larger than the median size. They are also defined as triangular functions peaking at the minimum and maximum values of the universe of discourse. An alternative to this approach could be to base the sets on the sizes of

the user-delineated shacks (see Section 5.1.1).

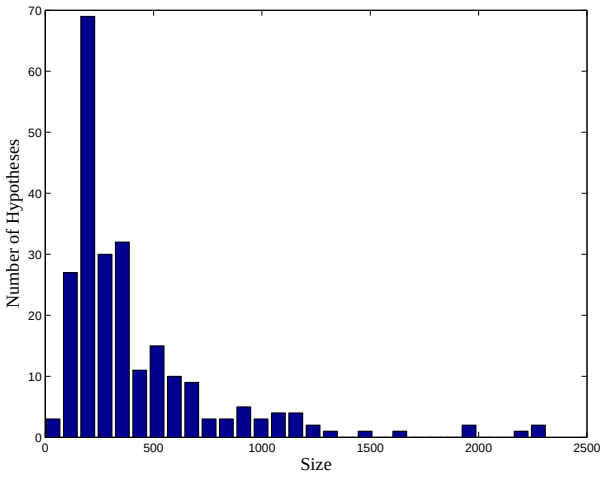


Figure 5.27: Histogram of hypotheses' sizes for the Marconi Beam image. A bin size of 80 is used.

The rectilinearity input variable is converted to a fuzzy value using the fuzzy sets in Figure 5.25b. These sets, defined by S-, pi-, and Z-curves, are symmetrical about a value of 0.5 dividing the universe of discourse into "Low", "Medium" and "High" values.

The two fuzzy sets for the compactness variable are given in Figure 5.25c. In defining these sets, it is possible to make use of domain-specific knowledge regarding the dimensions of a shack or building. Remember that compactness is defined as:

$$\mathscr{C}(P) = \frac{4\pi\mathscr{A}(P)}{(\mathscr{P}_e(P))^2} \tag{5.12}$$

Given the predominant type of shack being detected, it is useful to consider the case when the polygon, P, is a rectangle, with a length of l and a width of w. Expressing the width as a proportion of the length using $w = kl$ and substituting into Equation 5.12 gives:

$$\begin{aligned} \mathscr{C}(P) &= \frac{4\pi(l)(kl)}{(2(l+kl))^2} \\ &= \frac{\pi k}{k^2+2k+1} \end{aligned} \tag{5.13}$$

Equation 5.13 has a maximum value of 0.785 for P being a square ($k = 1$). As the sides of the rectangle grow more disproportionate so the compactness value decreases. The compactness value corresponding to a square is significant because shacks are unlikely to have higher compactness values (i.e. appear circular) according to the

assumed model. This knowledge is expressed in the construction of the fuzzy sets for *Compactness*. The compactness value for a square boundary marks the start of the drop-off of the linguistic value "Medium" which decays linearly to zero. Overlapping with this fuzzy set is the set for "High" compactness.

It is also possible to use Equation 5.13 for constructing fuzzy sets around the expected minimum value of compactness for a shack. For example, it may be assumed that the width-to-length ratio of rectangular shacks is unlikely to exceed $3:1$. Letting $k = 3$ or $\frac{1}{3}$ and substituting into Equation 5.13 gives a compactness value of 0.589. This value may then be used to demarcate fuzzy sets in a similar fashion. In practice, this knowledge was not used because it was found to offer little additional discriminatory power as fuzzy rules based on size tend to rule out long and narrow hypotheses in competing sets of hypotheses. It is also worth noting that deciding on a minimum value for compactness is not that straightforward because sometimes hypotheses identify portions of a shack roof but not the entire shack roof. These hypotheses are valuable as they may be grouped to delineate an entire roof in the grouping stage (Section 5.7) but it is not possible to be as confident about their expected dimensions.

The *Support* input variable incorporates the fuzzy sets "Low" and "High" which determine the degree of support based on the shadow-support score (Figure 5.25d). As with *Size*, the fuzzy sets are determined dynamically on a per-image basis. Even though the theoretical extent of the universe of discourse is $[0, 2]$, based on Equation 5.9, the actual range of scores for an image is restricted as scores at the extremes of the theoretical range are unlikely. Using fuzzy sets based on the actual range allows them to be more attuned to the image being dealt with. The linguistic value for "Low" is described by a Z-curve, starting at the smallest value of the actual range and decaying to the midpoint of the range. The linguistic value "High" is an S-shaped fuzzy set starting from a third of the way into the actual range and increasing until its upper limit. These curves are not symmetrical. They have been constructed to match the perceived quality of the support score, and in this case, a score at the midpoint of the range is felt to be a stronger indicator of the presence of a shack than it is a counter-indicator. Looking at Figure 5.25d, it is noticeable that there is no fuzzy set for the concept of "Medium" support. This is intentional as modelling the concept of "Medium" does not improve the output of the FIS, yet it results in a system having more rules and therefore greater complexity.

The single output variable of the FIS is depicted in Figure 5.26. In this case, there are five linguistic values ranging from "Very Unlikely" to "Very Likely" for shack or building *Likelihood*. *Likelihood* is an output variable taking the form of an aggregation

of fuzzy regions produced by the consequents of each of the fuzzy rules. This region is defuzzified using the common method of centroid defuzzification [134] to produce a value in the range $[0, 100]$ indicating the possibility that a hypothesis delineates a shack or building. The domain of the fuzzy set "Very Unlikely" is slightly wider than the widths of the other sets from their peaks to zero. This means, in effect, that "Very Unlikely" is weighted more heavily than other values. Rules which have a high degree of support for "Very Unlikely" will disproportionately lower the likelihood value. The use of this fuzzy set allows the presence/absence of a critical piece of evidence to dominate when negatively evaluating a hypothesis.

The fuzzy sets for the input and output variables have, in general, been determined empirically based on a subjective assessment of how well the base value (for example, the rectilinearity measure) satisfies the linguistic value (for example, "High") and the utility of these values in the rule base. This approach is common in developing fuzzy systems as these systems are tolerant of approximate fuzzy sets [134]. In some cases, such as *Size* and *Compactness*, the construction of the related fuzzy sets is strongly informed by polygon statistics and domain-specific knowledge. All of the above sets are used by fuzzy rules in evaluating the strength of a hypothesis.

The Fuzzy Rule Base

At the heart of a FIS is the fuzzy rule base. Fuzzy rules offer advantages over some other forms of evidence combination in that they are transparent — domain-specific knowledge is captured in an explicit form. Furthermore, they are based on natural language and are therefore easy to understand and modify.

The fuzzy rule base is given in Table 5.3. The rules are numbered for easy reference. The AND fuzzy operator is used to determine the strength of the antecedent for each of the rules involving two input variables. For instance, Rule 1, should be read as follows:

> If *Size* is Large AND *Support* is High then *Likelihood* is Very Likely.

The conjunction of two clauses is implemented using the standard *minimum* method. The strength of the antecedent clause is used to reshape the fuzzy set in the rule's consequent. This is termed "implication" and here, again, the minimum method is used. Finally, all of the consequent fuzzy sets are aggregated into a single set. The aggregation method used is simply to *sum* the consequent fuzzy sets. This

Rule	Input Variables				Output Variable	Evidence
	Size	Rectilinearity	Compactness	Support	Likelihood	
1	Large			High	Very Likely	Positive
2	Large			Low	Very Unlikely	Negative
3	Large	High			Likely	Positive
4	Large	Medium			Maybe	Neutral
5	Large	Low			Very Unlikely	Negative
6	Medium			High	Maybe	Neutral
7	Medium			Low	Maybe	Neutral
8	Medium	High			Maybe	Neutral
9	Medium	Medium			Maybe	Neutral
10	Medium	Low			Unlikely	Negative
11	Small			High	Maybe	Neutral
12	Small			Low	Unlikely	Negative
13	Small	High			Maybe	Neutral
14	Small	Medium			Maybe	Neutral
15	Small	Low			Unlikely	Negative
16			High		Very Unlikely	Negative
17			Medium		Maybe	Neutral

Table 5.3: Fuzzy rules grouped by *Size* value, then by decreasing *Support*/*Rectilinearity* value.

aggregation method allows evidence to be accumulated by having each rule contribute to the final solution [134]. The aggregated fuzzy set may have membership values exceeding one. This, however, is not an issue as centroid defuzzification takes place as normal.

From Table 5.3 it is evident that Rules 1–15 involve the *Size* variable. *Size* is used to govern which rules come into play during the evaluation of a hypothesis. The *minimum* of the degree of membership in the *Size* fuzzy set and the degree of membership in the additional variable (*Rectilinearity* or *Support*) forms the support for the rule. Thus, if a hypothesis is, for example, small, then the degree of support for all rules involving large-sized hypotheses will be zero, and the support for rules involving medium-sized hypotheses will be minor. Rules 16–17 are the only two rules that are based on *Compactness* and are independent of *Size*.

The rules are classified according to whether they strengthen, weaken or are neutral regarding the possibility of a hypothesis. This classification is made based on the

value of the output variable of the rule. Rules producing *Likelihood* values of "Very Likely" and "Likely" are regarded as providing positive evidence for a hypothesis, while rules producing values of "Very Unlikely" and "Unlikely" supply negative evidence. Rules that produce a fuzzy set output with a value of "Maybe" are seen to be neutral.

Rules 1–5 are concerned with large hypotheses. They intuitively model the expectation that the more rectilinear a hypothesis is and the more support it enjoys, the greater its likelihood. However, it is instructive to discuss the actual *Likelihood* values that are used. Comparing Rules 1 and 3 shows that *Support* is deemed to be a better indicator of likelihood than rectilinearity for large hypotheses because a high support value results in a *Likelihood* of "Very Likely", while for *Rectilinearity* a high value results in a less strong *Likelihood* of "Likely". In other words, shadow support along a long roof-shadow boundary is highly significant.

It is important to select a large hypothesis from a competing set only if it has a good chance of being correct. Choosing a large, *incorrect* hypothesis results in gross errors (based on both visual inspection and detection and quality metrics) when compared to incorrect small and medium hypotheses. Therefore, a conservative assessment of large hypotheses takes place in which an attempt is made to boost their chances of selection only if they have a large amount of evidence in their favour. Conversely, if they lack such evidence, their likelihood value is heavily penalised. This is expressed in the consequents for Rules 1, 2 and 5 which use values at the extremes of the *Likelihood* variable's range. For example, a large hypothesis with high support is "Very Likely"; however, if the support is low then it is at the opposite extreme of the range, being "Very Unlikely".

For medium-sized hypotheses, far less clear-cut decisions can be made based on the available evidence. This is due to several factors. As discussed earlier, the shadow-support score tends to favour hypotheses which are smaller in size. Given a large-sized structure with generally good shadow support, some of the smaller hypotheses in the competing set, which do not correspond to the entire roof but only to a portion, are likely to have higher support values than the larger hypothesis which offers more complete roof coverage. This is because there is a greater chance of smaller hypotheses being supported by unoccluded shadow than larger hypotheses. This lowers the confidence that can be gained from high support values when dealing with medium-sized hypotheses and therefore, the consequent of Rule 6 is chosen to be "Maybe". Rule 6 contrasts sharply with Rule 1, which also deals with high support values but in the context of large sizes. This highlights how evidence is

interpreted differently depending on the context.

On the flip-side, medium-sized hypotheses may have low support because the supporting shadow is occluded. However, this may occur in a situation where the actual structure is medium-sized, and the hypotheses, which are medium-sized with respect to all hypotheses, are, in fact, the largest hypotheses in the competing set. In such cases, low support is not necessarily an indicator of a poor hypothesis. This is reflected in Rule 7 in that the likelihood, even with low support, is still "Maybe". In effect, the support score offers no discriminatory power for medium-sized hypotheses as the likelihood value is the same irrespective of whether the support is high or low.

Rules 8–10 deal with medium-sized hypotheses and rectilinearity. As with rules dealing with support for medium-sized hypotheses, the evidence provided by different grades of rectilinearity is less conclusive than that for large hypotheses. However, if the rectilinearity is low the hypothesis is deemed "Unlikely".

Globally small hypotheses, more often than not, correspond to parts of roofs than to complete roofs. In cases where small hypotheses are competing against larger hypotheses, they will only be chosen if the larger hypotheses do not fit the model well. The fuzzy rules are tailored to select the small hypothesis with the most support and medium to high rectilinearity. Therefore, these rules encode the assumption that parts of shack roofs, in addition to the entire roof, will have a fairly rectilinear shape. This may not always be the case but, on the whole, seems to provide satisfactory results (see Chapter 6). On balance, medium hypotheses are selected over small hypotheses as there are more rules accumulating negative evidence for small hypotheses than there are for medium hypotheses.

Rules 16 is designed to exclude overly-compact, that is, roundish boundaries from being chosen. If the boundary shape is not overly-compact then Rule 17, which provides neutral evidence, comes into play.

Table 5.4 presents the fuzzy rules organised in different fashion. Here they are grouped by the same *Support* or *Rectilinearity* value and then by decreasing *Size* values. This organisation reveals how the interpretation of the support score and rectilinearity measure varies depending on the size of the hypothesis under consideration. As can be seen, a high support/rectilinearity score does not always imply strong likelihood ("Very Likely" or "Likely") — in many cases it provides neutral evidence. However, low support/rectilinearity scores generally result in likelihoods of "Very Unlikely" or "Unlikely". A "Medium" value for *Rectilinearity* stands out for its consistent interpretation. It provides neutral evidence irrespective of the size being dealt with

Rule	Input Variables				Output Variable	Evidence
	Size	Rectilinearity	Compactness	Support	Likelihood	
1	Large			High	Very Likely	Positive
6	Medium			High	Maybe	Neutral
11	Small			High	Maybe	Neutral
2	Large			Low	Very Unlikely	Negative
7	Medium			Low	Maybe	Neutral
12	Small			Low	Unlikely	Negative
3	Large	High			Likely	Positive
8	Medium	High			Maybe	Neutral
13	Small	High			Maybe	Neutral
4	Large	Medium			Maybe	Neutral
9	Medium	Medium			Maybe	Neutral
14	Small	Medium			Maybe	Neutral
5	Large	Low			Very Unlikely	Negative
10	Medium	Low			Unlikely	Negative
15	Small	Low			Unlikely	Negative
16			High		Very Unlikely	Negative
17			Medium		Maybe	Neutral

Table 5.4: Fuzzy rules grouped by *Support*/*Rectilinearity* value, then by decreasing *Size* value.

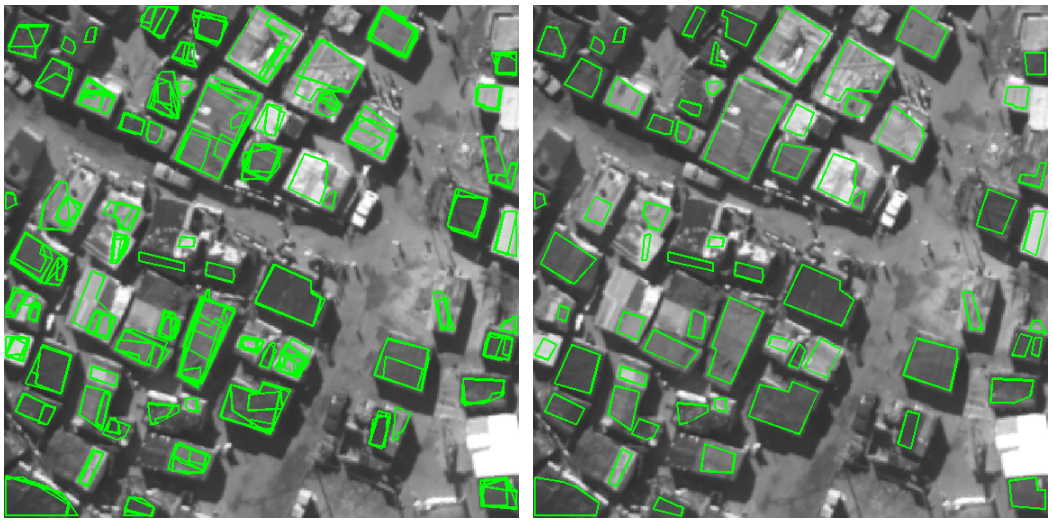
(Rules 4, 9 and 14).

For completeness, the exact structure and membership function parameters of the FIS for the Marconi Beam test image used in this chapter are given in Appendix C.

5.6.4 Results of Hypotheses Selection

The fuzzy rules are applied to each and every hypothesis to produce a likelihood score. The likelihoods for hypotheses in competing sets are compared and the hypothesis with the highest likelihood is selected. Figure 5.28 presents the results of hypotheses selection on the Marconi Beam image. The images illustrate that larger hypotheses are generally selected over smaller hypotheses in competing sets. However, this is not always the case. A large, erroneous hypothesis is visible corresponding to a shack on the left-hand side of Figure 5.28a, about a third of the way down from the top. Although this hypothesis is the largest in the competing set and is fairly rectilinear,

it lacks strong shadow support and therefore its likelihood is reduced. Figure 5.28b demonstrates that the fuzzy rules evaluate one of the smaller, and better fitting, hypotheses in the set as being more likely.



(a) Conflicting (overlapping) shadow-verified hypotheses.

(b) Hypotheses selected from conflicting sets.

Figure 5.28: Resolving conflicting sets of hypotheses for the Marconi Beam image.

The fuzzy rules used do not simply sum evidential support for a hypothesis (small hypotheses would mostly be chosen if this were the case) but introduce a measured bias towards larger boundaries, reflecting knowledge about the application domain, the hypothesis formation process, and the derived attributes.

Each of the competing hypotheses for a shack have been extracted from different scales in the scale-space. Referring back to Section 5.2.5 and, specifically, to Figure 5.9, the hypothesis selection process can be understood as forming a judgement on the optimal scale of appearance of a shack.

In [130, 131] a fuzzy inference system is also used for the selection of competing hypotheses while in [132] fuzzy logic is used for line segment grouping (hypothesis formation). Both of these systems use fuzzy logic within the context of an edge-based strategy. The use of fuzzy logic differs in this system in that it is used for hypothesis selection based on model conformance within the context of a region-based strategy. Contextual evidence for a hypothesis is incorporated through rules involving the shadow-support score.

5.7 Grouping

Anisotropic diffusion preserves strong edges over scale. Ideally such edges would only delineate shack boundaries. However, sometimes the contrast between two different roofing materials on the same shack is large enough to produce a strong internal edge which prevents the regions, corresponding to these different materials, from merging. This results in the shack roof being fragmented into different hypotheses. In order to recover the entire roof, fragmented hypotheses belonging to the same roof are identified. Different grouping combinations are considered, a composite boundary for each combination is formed and the best boundary selected. All of these topics are discussed in the following subsections.

5.7.1 Hypothesis Classification

In other work fragmented hypotheses are grouped by relating several fragments to a shared shadow, and fusing them into a complete structure [45]. Roof-shadow boundaries are determined and the endpoints of the boundaries are back projected a distance proportional to the shadow length along the sun vector. This forms a region of interest which is used to identify candidates for grouping. This approach is less viable in situations where shadows from many different shacks merge and back projecting the roof-shadow boundary will result in a region of interest encompassing many individual shack hypotheses.

The approach adopted here is to classify all the hypotheses extracted from the image (Figure 5.16b) and then group those which are related. The classification process is based on the idea of *support*. Hypotheses, as has already been shown, may have shadow support. They may also have *hypothesis* support. Hypothesis support is determined in a very similar manner to shadow support. The sun vector is translated along the roof-shadow boundary of the hypothesis in question and is sampled at regular intervals. Instead of only shadow samples being of interest, samples which fall in adjacent hypotheses are taken into account as well. The equation used earlier (Equation 5.9) for calculating shadow support is also used for calculating hypothesis support.

The assumption behind using hypothesis support to group fragments is that a fragment, adjacent to another shadow-verified fragment belonging to the same shack, will not cast shadow but should have a degree of hypothesis support in the sun vector direction. It is assumed that hypotheses correspond to structures of fairly

equal height, which is defensible when dealing with shacks. If this assumption is not made then the possibility that a hypothesis corresponds to a piece of raised roof substructure that is casting shadow onto other parts of the roof has to be dealt with, which complicates matters.

The degree of support will vary depending on the orientations of the hypotheses involved with respect to the sun vector. This is illustrated in Figure 5.29 where the hypothesis support for hypothesis A by hypothesis B is depicted. In Figure 5.29a the supported hypothesis, A, has a large amount of support from B and it is worthwhile constructing a composite boundary. Figure 5.29b depicts a case where the amount of support is partial as the sun vectors along A's roof-shadow boundary only partially overlap with B. Adequate support is determined by using an empirical threshold (see Appendix B). Figure 5.29c depicts a situation where the two hypotheses are oriented perpendicular to the sun vector, resulting in a hypothesis support score of zero. These hypotheses, which may correspond to the same shack roof, will not be considered candidates for grouping and will remain separated in the final interpretation.

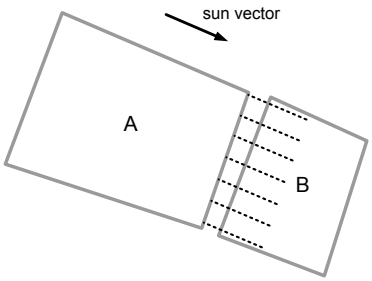
Hypothesis classification which is based on shadow and hypothesis support is used to determine which hypotheses will be grouped. The overall classification scheme is given in Figure 5.30, the end product of which is subsets of all of the hypotheses which form the input to the grouping process. These sets are described in more detail in the following subsections. The support equation, presented earlier, is used in forming some of the sets, and is repeated here for convenience.

$$\text{Support} = \left(\frac{S_{det} - S_{non}}{S_{tot}} + 1 \right) * \frac{RS_{det}}{RS_{tot}} \tag{5.9}$$

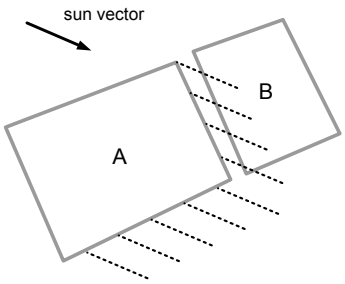
Depending on the circumstances, the samples being classified as detections and non-detections vary. This is described in more detail below. In all applications of Equation 5.9, the penalty factor only comes into play if neither shadow nor a hypothesis is sampled along a line segment of the roof-shadow boundary.

Shadow-Verified Hypotheses

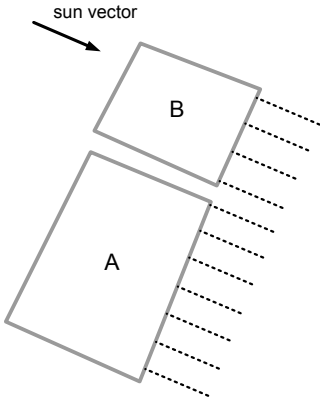
The generation of this set of hypotheses has already been described in Section 5.4.



(a) Longitudinal axis of the hypothesis grouping lies parallel to the sun vector, giving A a large support score.



(b) Longitudinal axis of the hypothesis grouping lies at an angle to the sun vector, giving A an intermediate support score.



(c) Longitudinal axis of the hypothesis grouping lies perpendicular to the sun vector, giving A a support score of zero.

Figure 5.29: Hypothesis support for differing orientations of hypotheses A and B. Hypothesis A is the supported hypothesis, hypothesis B is the supporting hypothesis. Hypotheses are drawn in thick gray lines; translations of the sun vector along the roof-shadow boundary are drawn in dashed, black lines.

Non-Conflicting, Shadow-Verified Hypotheses

Using fuzzy rules to eliminate conflicting, shadow-verified hypotheses has been detailed in Section 5.6.

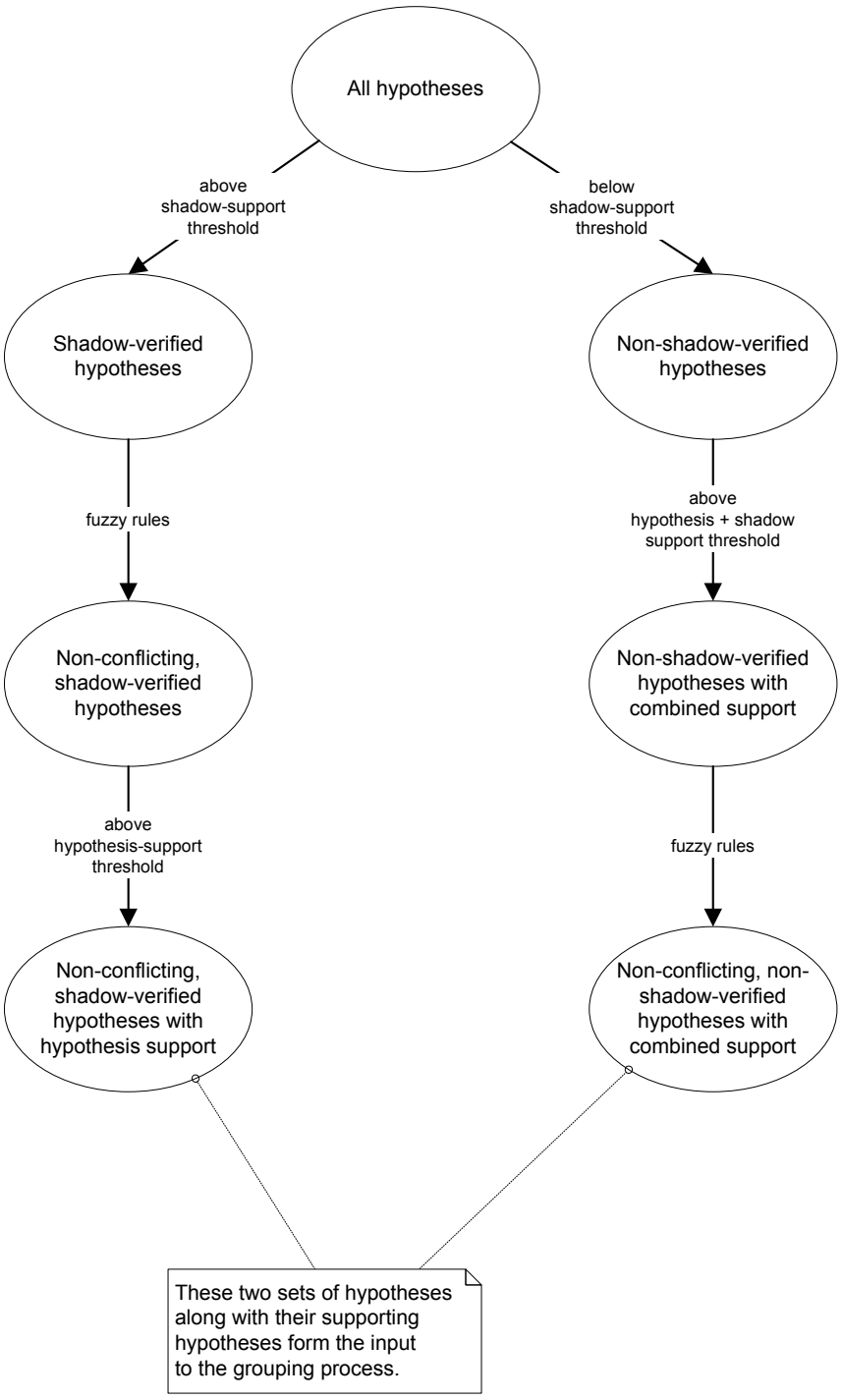


Figure 5.30: Classification of hypotheses. Ellipses represent sets of hypotheses. Arrows indicate a relation between two sets of hypotheses: the set at the end of each arrow being a *subset* of the set of hypotheses described at the start of arrow. Each of the arrows are annotated indicating the process by which the subset pointed to is formed. The diagram should be read from top to bottom beginning with the set of all extracted hypotheses.

Non-Conflicting, Shadow-Verified Hypotheses with Hypothesis Support

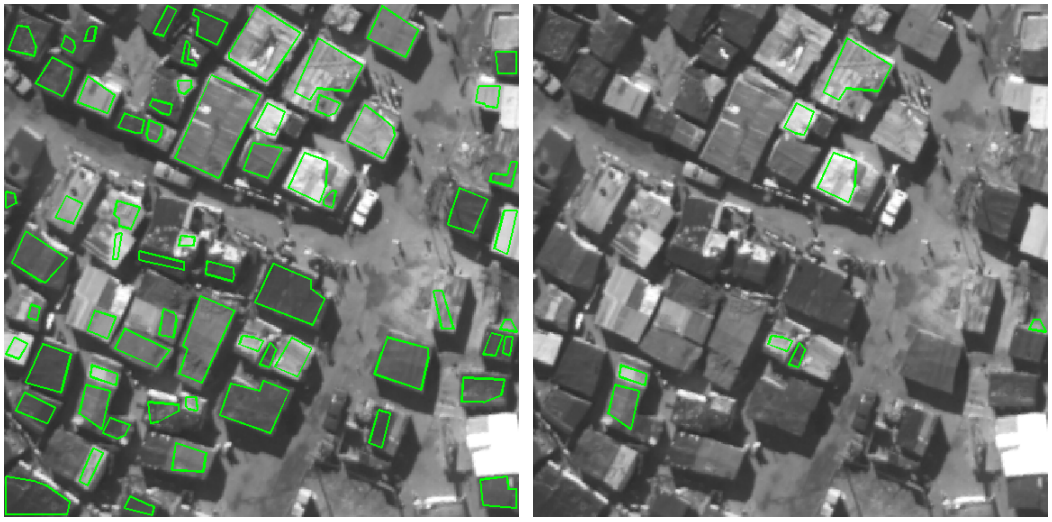
A shack roof exhibiting strong contrast between roof materials may be split into two or more hypothesis fragments, all of which are shadow-verified. In producing this set, the hypothesis support for each shadow-verified hypothesis in the enclosing set is calculated using the support equation. Samples which are classified as detections (S_{det}) are those lying within shadow-verified hypotheses. All remaining samples, that is, shadow samples and samples at which neither shadow nor a hypothesis is found are deemed to be non-detections (S_{non}):

$$S_{det} = S_{hypothesis}; \quad S_{non} = S_{shadow} + S_{no-shadow-no-hypothesis} \tag{5.14}$$

Hypothesis support is calculated for hypotheses that have *already* been shadow-verified. Therefore, shadow is not counted as a detection but as a non-detection and only hypothesis samples add to the support score. The sample sequence along the sun vector is important, as in the case when determining shadow support. Starting from the roof-shadow boundary, all non-detections are counted as well as the samples belonging to the first hypothesis which is encountered. No further samples are counted beyond the first hypothesis.

The hypothesis samples of the entire set may all lie in a single supporting hypothesis or they may lie in multiple supporting hypotheses. A relation is formed between the hypothesis under consideration, the *supported* hypothesis, and the *supporting* hypotheses. At least 10% of the total samples have to lie within a single supporting hypothesis for the support relation to be created. This minimum requirement helps to prevent spurious relations from forming. There are no other restrictions on the support relation, which means that at the completion of the classification process a hypothesis may end up supporting multiple hypotheses and being supported by multiple hypotheses (a many-to-many relationship).

Figure 5.31 illustrates the shadow-verified hypotheses which have sufficient hypothesis support.



(a) Shadow-verified hypotheses after conflict resolution.

(b) Shadow-verified hypotheses with hypothesis support.

Figure 5.31: Identification of shadow-verified hypotheses with hypothesis support in the Marconi Beam image.

Non-Shadow-Verified Hypotheses

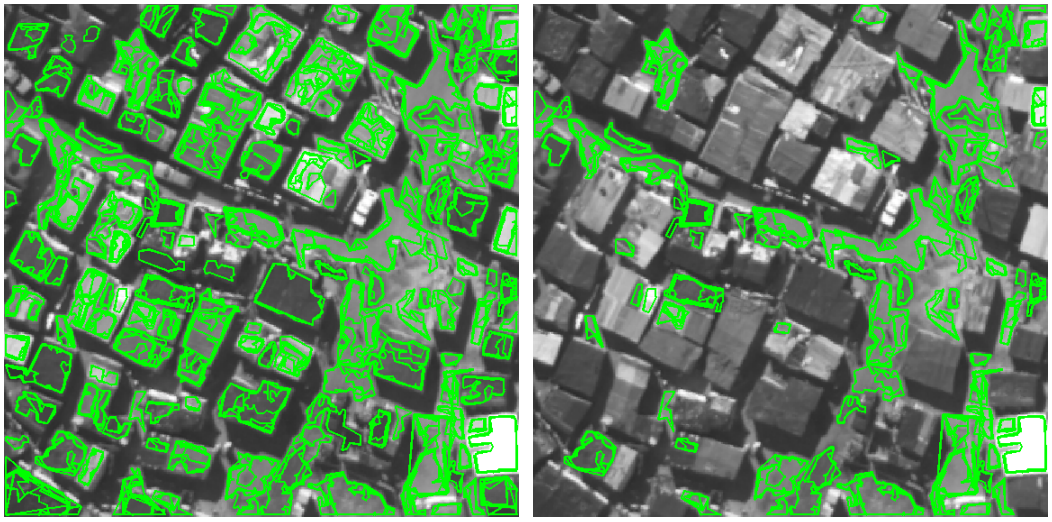
At the top of the righthand branch of Figure 5.30, the ellipse representing a set of non-shadow verified hypotheses is drawn. The shadow-support score for these hypotheses falls below the shadow-support threshold. This set of hypotheses is shown superimposed on the source image in Figure 5.32b.

Non-Shadow-Verified Hypotheses with Hypothesis Support

Fragmented hypotheses may correspond to roof fragments which do not have any, or have very little, shadow support if they are located away from the roof-shadow boundary. It has already been established that these hypotheses do not have sufficient shadow support to be shadow-verified, but they may have sufficient *combined* support from both shadow and other hypotheses to merit being considered for grouping.

In using Equation 5.9 to calculate combined support, detections and non-detections are defined as:

$$S_{det} = S_{hypothesis} + S_{shadow}; \quad S_{non} = S_{no-shadow-no-hypothesis} \tag{5.15}$$



(a) All hypotheses extracted from the source image.

(b) Non-shadow-verified hypotheses (hypotheses with insufficient shadow support).

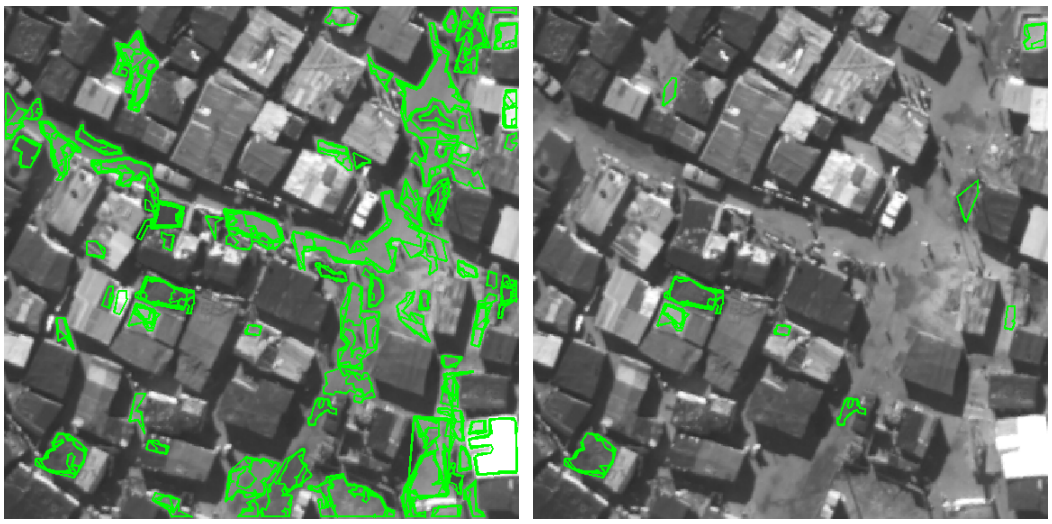
Figure 5.32: Non-shadow-verified hypotheses for the Marconi Beam image. Hypotheses from all stack levels are superimposed on the source image.

For a non-shadow-verified hypothesis being evaluated, any samples which lie within the set of non-conflicting, shadow-verified hypotheses are counted as hypothesis samples, $S_{hypothesis}$. Moving in the direction of the sun vector, all non-detections are counted as well as the samples belonging to the first shadow or hypothesis sampled. Beyond this no further samples are counted.

Figure 5.33a illustrates all the non-shadow-verified hypotheses. As is evident in Figure 5.33b, very few of these have sufficient combined support. This is largely because there are few non-shadow-verified hypotheses corresponding to roof fragments. Comparing the two images in Figure 5.33, some sets of polygons which do partially overlap shack roofs have been completely filtered out (false negatives). This is because these sets are isolated having no detected shadow or shadow-verified hypotheses nearby. Some false positives are also present in the right-hand image.

Non-Conflicting, Non-Shadow-Verified Hypotheses with Combined Support

As with the set of shadow-verified hypotheses, the set of non-shadow-verified hypotheses with combined support includes hypotheses from many different stack levels which may overlap with one another. Overlapping hypotheses are viewed as



(a) All non-shadow-verified hypotheses.

(b) Non-shadow-verified hypotheses with sufficient combined support.

Figure 5.33: Identification of non-shadow-verified hypotheses with combined shadow and hypothesis support in the Marconi Beam image. Hypotheses from all stack levels are superimposed on the source image.

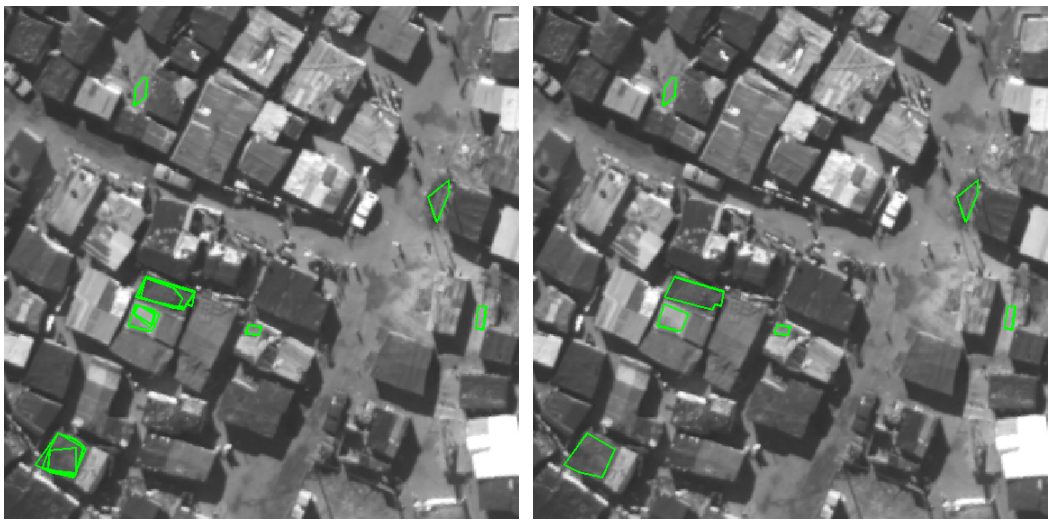
being in contention and, as before, a FIS is used to select the best hypothesis. The FIS structure and parameters are identical to that described in Section 5.6.3 and Appendix C.

Figure 5.34a presents the regularised boundaries for non-shadow-verified hypotheses having sufficient combined support, while Figure 5.34b gives the result of hypothesis selection from these competing hypotheses.

In classifying hypotheses three different thresholds have been used:

- the shadow-support threshold,
- the hypothesis-support threshold and,
- the combined shadow and hypothesis support threshold.

In this system the shadow-support and hypothesis-support thresholds are identical. All three threshold values are given in Appendix B and they are identical for all the images tested (system parameters).



(a) Model-boundaries for non-shadow-verified hypotheses with sufficient combined support.

(b) Hypotheses selected from conflicting sets.

Figure 5.34: Resolving conflicting sets of non-shadow-verified hypotheses for the Marconi Beam image.

5.7.2 Grouping Combinations

The two sets of hypotheses shown at the bottom of Figure 5.30, along with their supporting hypotheses, are subject to the grouping procedure. Each hypothesis in each set is linked to the others through support and supporting relations. All hypotheses which are linked together through these relations form a *group*. Each hypothesis within a group is termed a *grouping candidate*. Each group is composed of n shadow-verified hypotheses (including associated supporting hypotheses which are always shadow-verified) and m non-shadow-verified hypotheses. Therefore, a group consists of $n + m$ hypotheses.

Different combinations of grouping candidates exist, termed *grouping combinations*. The composite boundary or polygon which results from a particular grouping combination is known as the *grouped boundary*. Not all grouping combinations are permissible. The rules governing permissible combinations are:

- Each combination must be composed of a minimum of two grouping candidates.
- Each combination must include at least one shadow-verified hypothesis.

Within these constraints all possible combinations for a group are determined, the total number being given by either Equation 5.16 or 5.17.

$$C_G = \sum_{k=1}^{m} \frac{m!}{k!(m-k)!} \qquad \text{for } n = 1 \qquad (5.16)$$

$$C_G = \sum_{k=2}^{n} \frac{n!}{k!(n-k)!} + \left(\sum_{k=1}^{m} \frac{m!}{k!(m-k)!} \right) \left(\sum_{k=1}^{n} \frac{n!}{k!(n-k)!} \right) \qquad \text{for } n \geq 2 \qquad (5.17)$$

where

C_G = number of permissible combinations for a group
m = number of non-shadow-verified hypotheses in the group
n = number of shadow-verified hypotheses in the group
k = number of grouping candidates

Equation 5.16 determines the number of permissable combinations for the case where there is a single shadow-verified hypothesis in the group. This hypothesis will form part of each grouping combination, allowing the total number of combinations to be calculated by considering only the non-shadow-verified grouping candidates. Equation 5.16 gives all possible combinations of these candidates for all possible group sizes. Note, group sizes of one are included as each non-shadow-verified hypothesis is paired with the single shadow-verified hypothesis to meet the constraints above.

Equation 5.17 handles the case when there are two or more shadow-verified grouping candidates. The first term gives all the possible combinations that exist for groupings of two or more shadow-verified hypotheses. The second term gives the total number of combinations that exist for groupings which include both shadow-verified and non-shadow-verified candidates.

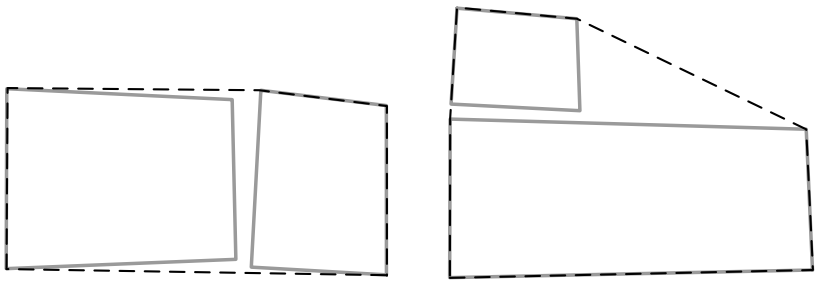
The total number of combinations to be evaluated grows rapidly as the size of the group increases. Potentially, this could be problematic in terms of the computational effort required to evaluate a large number of combinations. In practice, however, group sizes remain small, with most groups consisting of four or fewer grouping candidates.

The combination constraints listed above are conservative with regard to non-shadow-verified grouping candidates. These candidates will only be included in the final interpretation of the scene if they can be successfully grouped with shadow-verified hypotheses. The reason for this is that for non-shadow-verified hypotheses no direct image evidence exists — only weaker, inferred evidence in that these hypotheses are located in the reverse direction of the sun vector with respect to shadow-casting hypotheses. Such candidates may, for example, correspond to sand patches which happen to be appropriately located. To further prevent erroneous groupings, verification of the grouped boundary is performed. This is described in Section 5.7.4.

5.7.3 Forming a Grouped Boundary

A grouped boundary for a particular grouping combination is composed from the boundaries of the individual hypotheses involved. The grouped boundary is simply the convex hull of the individual boundaries or polygons. The convex hull, H, of a set of points, P, is defined as the smallest convex polygon for which each point in P is either on the boundary of H or in its interior. The vertices of the convex hull are a subset of the points P. P, in this scenario, refers to the vertices of the grouping candidates' boundaries that are being combined.

Two rectilinear, quadrilateral hypotheses in a "matched" configuration (their closest sides being fairly parallel, of the same length and not offset with respect to one another) will be grouped together into a larger, rectilinear boundary as in Figure 5.35a. However, two quadrilateral hypotheses which could potentially form a rectilinear L-shaped building will have a convex hull with a relatively low rectilinearity (see Figure 5.35b) and grouping will not take place as the grouping criterion, which is based on rectilinearity (see Equation 5.18), will not be met. This is a limitation of using the convex hull method to form the grouped boundary — L-shaped hypotheses can only be produced at model-driven simplification stage but not at the grouping stage.



(a) Rectilinear convex hull resulting from the grouping of two hypotheses in a "matched" configuration.

(b) Non-rectilinear convex hull resulting from the grouping of two hypotheses in an L-shaped configuration.

Figure 5.35: Grouping hypotheses using convex hulls. Hypotheses are drawn in thick gray lines; the convex hull is drawn in a thinner, dashed, black line.

5.7.4 Grouped Boundary Selection

For each set of grouping candidates a number of grouped boundaries are produced. Some of these boundaries may be the result of grouping incorrect non-shadow-verified hypotheses (hypotheses which do not correspond to roof fragments) with shadow-verified hypotheses. Additionally, grouped boundaries will overlap if a group contains three or more grouping candidates. Algorithm 5.2 is used both to filter out incorrectly grouped candidates and to eliminate grouping conflicts.

Input: G – set of grouping candidates
Output: S – set of selected grouped boundaries {set is initially empty}

1: $B \leftarrow$ the set of grouped boundaries for every permissible grouping combination of G
2: $k \leftarrow C_G$ {C_G, the number of boundaries in B, is given by Equation 5.16 or 5.17}
3: terminate $\leftarrow$ **false**
4: **while** $B \neq \phi$ **and** terminate is **false do**
5: $b \leftarrow B_i$ where i gives $max_{i \in [1,k]} \mathscr{R}(B_i)$ {b – grouped boundary with highest rectilinearity}
6: **if** acceptance criterion is satisfied **then** {see Equation 5.18 }
7: $B \leftarrow B - b, \quad S \leftarrow S \cup b$ {move b from set B to set S}
8: remove all boundaries from B that overlap with b
9: $k \leftarrow$ total remaining boundaries in B
10: **else**
11: terminate $\leftarrow$ **true** {no boundaries in B meet the criterion}
12: **end if**
13: **end while**

Algorithm 5.2: Algorithm for grouped boundary selection. Comments are included in braces.

In line 1 the set of all possible grouped boundaries is formed by finding convex hulls for each grouping combination. Boundary selection takes place in the "while" loop. Initially, the grouped boundary with the highest rectilinearity is considered. If this boundary meets the acceptance criterion, then it is selected and any other grouped boundaries that overlap with it are eliminated (line 8). This process repeats until the acceptance criterion is not met or there are no more grouped boundaries to consider. The acceptance criterion is given in Equation 5.18 and prescribes a minimum rectilinearity for the grouped hypothesis. This criterion ensures that the rectilinearity of the grouped boundary is not much less than the rectilinearity of any of the individual shadow-verified hypotheses of which the group is composed. In effect, this equation is used to verify grouping combinations. It is based entirely on rectilinearity, as this measure is an adequate discriminator between desirable and

undesirable groupings.

$$\mathscr{R}(b) \geq 0.75 \cdot (\max_{i \in [1,n]} \mathscr{R}(v_i)) \tag{5.18}$$

where

v_i = boundary of a particular shadow-verified hypothesis in the group being considered

b = a grouped boundary formed by taking the convex hull of shadow-verified and possibly non-shadow-verified hypotheses

n = total number of shadow-verified hypotheses in the group being considered

The output of the algorithm is the set, S, of selected grouped boundaries which replace individual grouping candidates. S may either contain one or more non-conflicting grouped boundaries or it may be an empty set, if none of the boundaries meet the acceptance criterion. A final overlap test is performed between the boundaries in S and the shadow-verified grouping candidates. All shadow-verified candidates are retained unless they have been grouped into larger hypotheses with sufficiently high rectilinearity. Grouped boundaries are illustrated in Figure 5.36. This figure demonstrates that the grouping procedure can successfully merge hypothesis fragments corresponding to the same shack roof.

5.8 Hypothesis Boundary Expansion

As mentioned earlier, the hypotheses' boundaries sit slightly within the interiors of the shack roofs. This originates with the segmentation process which identifies homogeneous regions in the image. These regions identify homogeneous roof interiors but do not extend right up to roof boundaries as this is where changes in intensity occur if a roof is distinct from its surroundings.

In order to improve roof coverage the following overarching assumptions are made:

1. A new, better localised boundary is to be found outside of the existing boundary, that is, expansion is required.
2. The new boundary closely matches the existing boundary in shape.

The following sections detail a boundary expansion technique which uses image edges and is based on the above assumptions. The existing boundaries are now termed

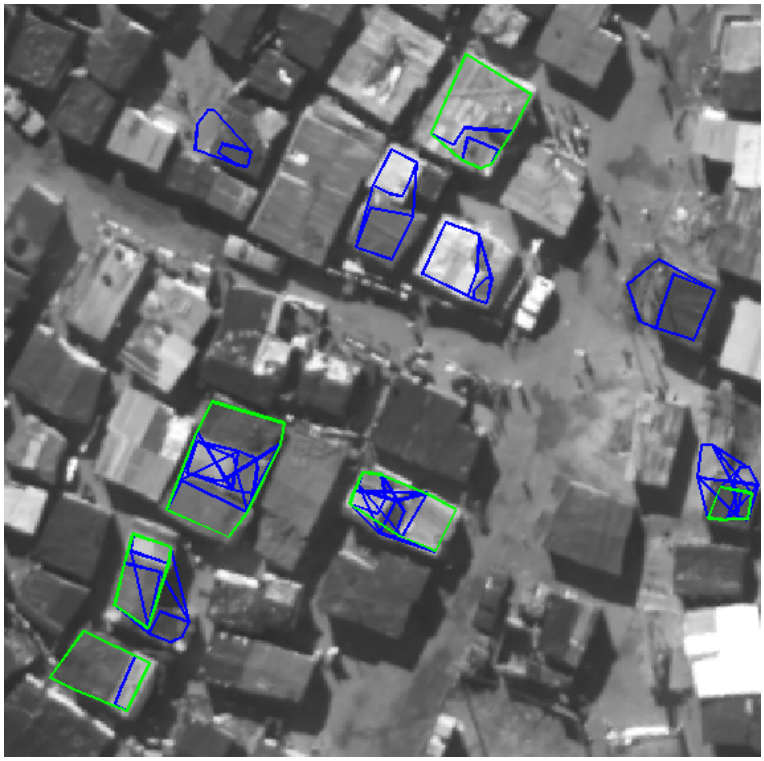


Figure 5.36: Grouping hypotheses for the Marconi Beam image. All permissable grouped boundaries are drawn in blue. The selected grouped boundaries are overlaid in green.

reference boundaries as expansion occurs with reference to these boundaries.

Work on perceptual grouping of edges for building detection [69, 70, 72, 36, 132] is related to this section. However, a significant difference is that, in the work presented here, an approximate boundary already exists prior to the use of edges.

In a more similar vein to this work is the approach used in [135]. Xie et al. [135] use a thresholded digital elevation model (DEM) image to create masks identifying 3D structures within an image. These masks are dilated to ensure that rooftops are completely covered. Edges are then found using a Canny edge detector [65] on an optical image of the scene, and only edges lying within the masks are retained for perceptual grouping.

5.8.1 Edge Detection, Straight Line Approximation and Filtering

Edges are detected on the source image (the image at stack level one) using the Canny edge detector [65]. The thresholds for the Canny edge detector are chosen (see Appendix B) so that relatively weak edges are found. Although this results in a

large total number of edges, this approach is feasible because the edges of interest are restricted to areas surrounding the reference boundaries and form a small subset of the total.

Edges of interest are identified through the following steps:

1. The Canny edge detector is applied producing an edge magnitude image.
2. All edge pixels lying in the interior of reference boundaries are removed.
3. The remaining edge pixels are linked together to form 8-connected chains of a minimum length. If a pixel chain branches, one branch is followed and then the algorithm returns to the junction and tracks the other branches.
4. Pixel chains are then approximated by straight lines. A simple split algorithm is used [136, p.196] wherein the initial approximation of the chain of edge pixels is a straight line joining the endpoints. This line is recursively subdivided at the point of maximum deviation until the maximum deviation of each of the approximating line segments is less than the allowable tolerance. These straight line approximations are simply referred to as "edges" from here on.
5. Search windows are formed for each reference boundary. The vertices of the search window polygon lie a specified distance outside of the reference boundary on the lines bisecting the inner angles of each reference boundary vertex.
6. All edge pixel chains (and their corresponding edges) which do not have at least one pixel within the narrow strip surrounding each reference boundary (the width of the strip is bounded by the search window) are removed. The remaining edge pixels and their edges are illustrated for a single hypothesis in Figure 5.37.
7. Edges are filtered according to additional constraints in order to try and eliminate edges which do not correspond to shack roof boundaries:
    - Each edge is associated with the boundary segment to which it is most closely aligned (parallel). A 30° difference in angle between the edge and its associated segment is allowed. If an edge is not aligned to within this tolerance then it is rejected. An unaligned edge, A, is depicted in Figure 5.38 while an aligned edge is shown as D.
    - Furthermore, each edge must lie partially or fully in the region to the left of its associated reference boundary segment (the reference boundary is directed clockwise). This region is bounded at each end by lines perpendicular to the associated segmented. This is shown in Figure 5.38 where edge B falls outside its respective region and is therefore, rejected.

- Finally, in the case of double edges where one edge overlaps, partially or fully, with another in their projection onto the reference boundary, the edge furthest from the reference boundary is removed (see Figure 5.38 C). This simple approach to dealing with double edges is effective when the edges forming a pair belong to two different hypotheses which happen to be in close proximity to one another. However, there are situations when an edge marking rooftop detail eliminates the true boundary edge. This situation is somewhat ameliorated by the fact that usually multiple fragmented edges mark the true boundary line so that if one true edge is eliminated others will remain to provide evidence for the position of the actual boundary. Alternatively, a more sophisticated approach to dealing with double edges can be used [48].

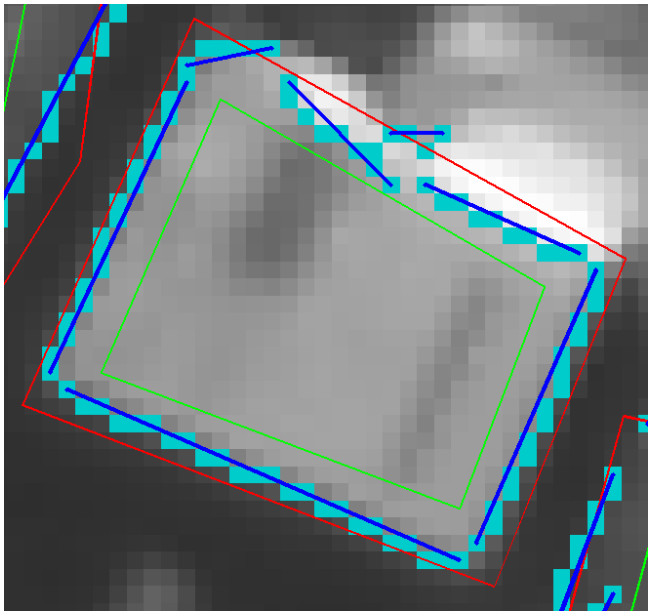


Figure 5.37: Edge pixels and straight line approximations surrounding a reference boundary. Edge pixels are coloured light blue; straight line approximations are drawn in dark blue; the reference boundary is in green and the search window boundary in red.

Using edge information facilitates an additional verification of the reference boundaries in which the shadow-support score for edges associated with roof-shadow boundary segments is calculated, and at least one edge has to have support exceeding a threshold. The edge shadow-support score is calculated in a similar fashion to support for hypotheses, using Equation 5.9, though, there is no penalty factor ($\frac{RS_{det}}{RS_{tot}}$). This verification acts as a final sanity check before boundary expansion and re-simplification occurs.

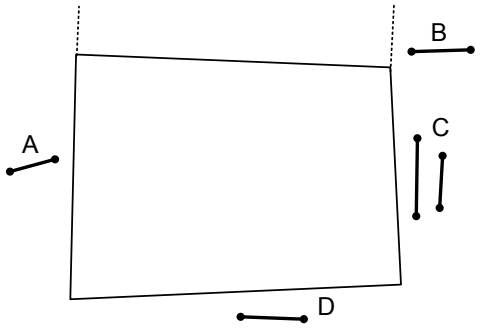


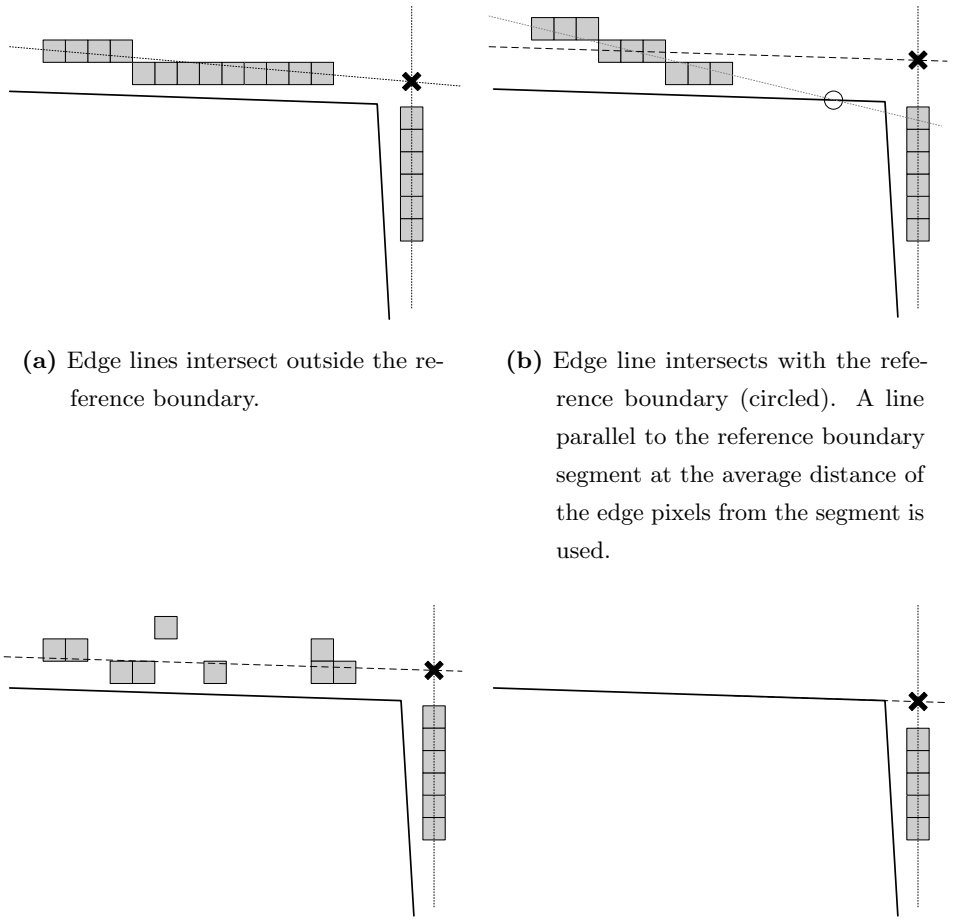
Figure 5.38: Filtering edges surrounding a reference boundary. A – unaligned edge, removed; B – edge not in region formed by associated boundary segment and dashed lines, removed; C – double edge, edge furthest from the associated boundary segment is removed; D – aligned edge, retained.

5.8.2 Vertex Creation for Expanded Boundaries

At this stage each reference boundary segment has zero or more aligned edges associated with it. These edges are used to generate new boundary points which lead to a better localised boundary. Recall, that up until this stage hypotheses' boundaries have undergone simplification through vertex removal. In order for them to expand outwards, new vertices have to be created. These vertices generally correspond to corners and are created by intersecting lines passing through the filtered edges.

The lines which are intersected are created in different ways depending on the edge evidence available. In the best scenario two edges exist and the lines which pass through them intersect outside the reference boundary (Figure 5.39a). Second to this is the situation where two edges exist but the edge line for one of the edges intersects with the associated boundary segment (Figure 5.39b). Here, the edge evidence slightly contradicts the expected shape based on the reference boundary. If the two lines are intersected, as in the previous situation, the resulting corner point may cause the final boundary to substantially cut across the corner of the reference boundary, violating the expansion and shape similarity assumptions. To prevent this from happening, a line parallel to the associated boundary segment at the average distance of the edge pixels from the segment is used for calculating intersection points. Figures 5.39c and 5.39d illustrate scenarios for increasingly degrading edge evidence. In the worst scenario, no edge evidence is available at all and a line collinear with the boundary segment is used.

All the edge lines associated with two neighbouring reference boundary segments are



(a) Edge lines intersect outside the reference boundary.

(b) Edge line intersects with the reference boundary (circled). A line parallel to the reference boundary segment at the average distance of the edge pixels from the segment is used.

(c) No edge associated with a reference boundary segment. A line parallel to the reference boundary segment at the average distance of the edge pixels from the segment is used.

(d) No edge and no edge pixels associated with a reference boundary segment. A line collinear with the reference boundary segment is used.

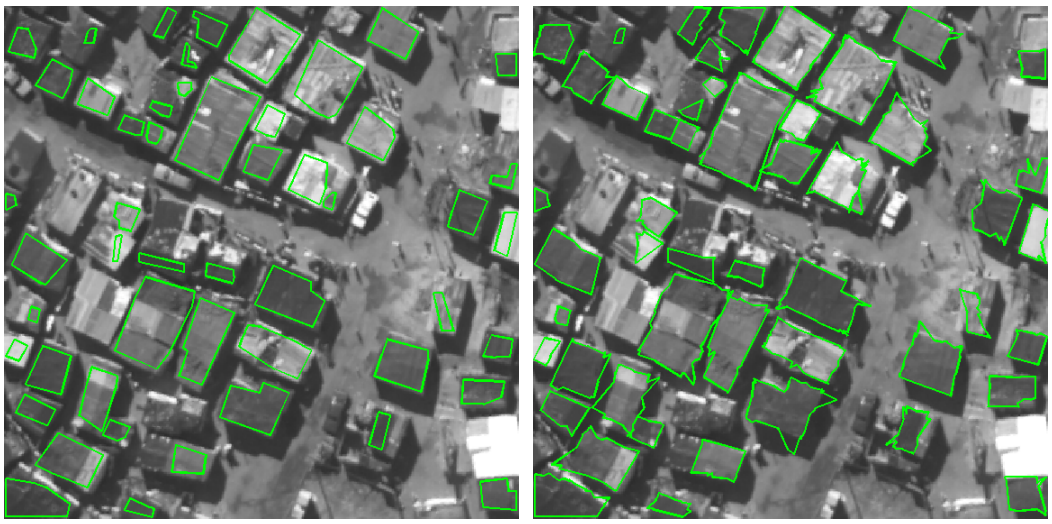
Figure 5.39: Intersections for different edge evidence scenarios. The reference boundary is given as a solid black line; edge pixels are indicated by grey blocks; edge lines are drawn as dotted lines; lines parallel to boundary segments are drawn as dashed lines; intersections are marked by crosses.

intersected to produce new points which replace the reference boundary. Additionally, the end points of each edge, which help to localise the positions of boundary segments and occasionally directly identify corners, are added to the new boundary. All of the new boundary points are ordered by performing a radial sweep from the reference boundary's centroid. The point sequence is based on the order in which each point is encountered in the sweep. After this is completed each boundary consists of:

- vertices produced as a result of edge line intersections, and

- vertices corresponding to the endpoints of the filtered edges surrounding the boundary.

Figure 5.40 illustrates the effect of expanding the reference boundaries. As can be seen the expanded boundaries are quite jagged, especially at the corners of hypotheses where multiple corner points have been generated. In some cases substantial deviations with respect to the true boundaries occur. This is often due to the presence of spurious edges which have not been filtered out.



(a) Edge-verified boundaries for hypotheses post-grouping. These boundaries form *reference* boundaries.

(b) Expanded reference boundaries.

Figure 5.40: Expanding reference boundaries for the Marconi Beam image.

Spurious edges which deviate sharply in alignment from their associated reference boundary segment could be filtered out by reducing the edge tolerance angle (set to 30°). However, the canonical orientation of the reference boundary may be skewed with respect to the true boundary. In order to correct the skew, it is necessary to be more lenient when filtering edges. Spurious edges may also exist which are well aligned with the reference boundary but are located too near/far from the boundary. All in all, it is difficult to distinguish true edges from false edges so the approach taken here is to let each and every edge within tolerance play a role in expanding the boundary, given that

- the number and quality (see the next section) of true edges associated with a boundary segment is often greater than that of false edges and hence, more evidence for true corner points is provided.

- the global shape constraints that are brought to bear when re-regularising the expanded boundary favour good corner points by removing perturbations which lower the rectilinearity and compactness of the boundary.

5.8.3 Model-Driven Simplification of Expanded Boundaries

The boundaries in Figure 5.40b are re-simplified in order to improve their localisation and appearance.

The first application of model-driven-simplification (Section 5.5) did not involve the use of image evidence in any form. At this stage of the detection process, however, pertinent edge evidence is available and can be taken advantage of during the simplification process. The same algorithm (Algorithm 5.1) is used as before but the expanded boundary, EB, forms the input instead of the boundary, SB. The objective function (line 9) is changed to include an additional term which factors in the support for each boundary point based on the edge(s) related to it. The new formulation is:

$$O = \mathscr{R}(EB) + \mathscr{C}(EB) + \sum_{k=1}^{i} P_k \tag{5.19}$$

where

O = objective function for expanded boundaries
$\mathscr{R}(EB)$ = rectilinearity measure for the expanded boundary, EB
$\mathscr{C}(EB)$ = compactness measure for EB
P_k = support for each point (or vertex) of EB
i = total number of vertices of EB

The point support P is determined as follows:

$$P = (l_1 - t_1) + (l_2 - t_2) \tag{5.20}$$

where

P = point support for an individual point
l_1 = length of first edge, l_1, participating in the intersection
t_1 = length from the nearest endpoint of edge, l_1, to the intersection point
l_2, t_2 = as above for the second edge participating in intersection

Equation 5.20 expresses the idea that an intersection point which is formed by longer edges and with shorter edge gaps to the intersection is more valuable. Point support is calculated for all intersection points in the expanded boundary. Points which

correspond to edge endpoints, and also form part of the expanded boundary, have their support values set to the average intersection point support. In other words, edge endpoints do not overtly influence the simplification process but do have more of an effect than intersection points with weak support.

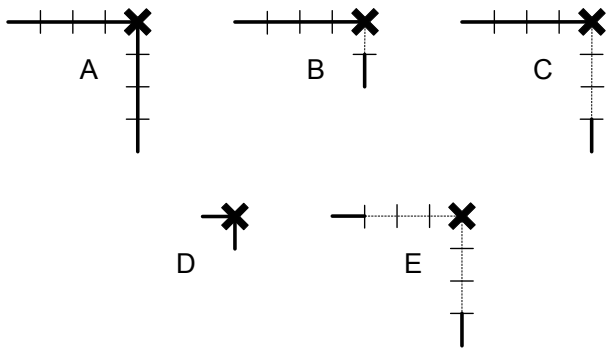
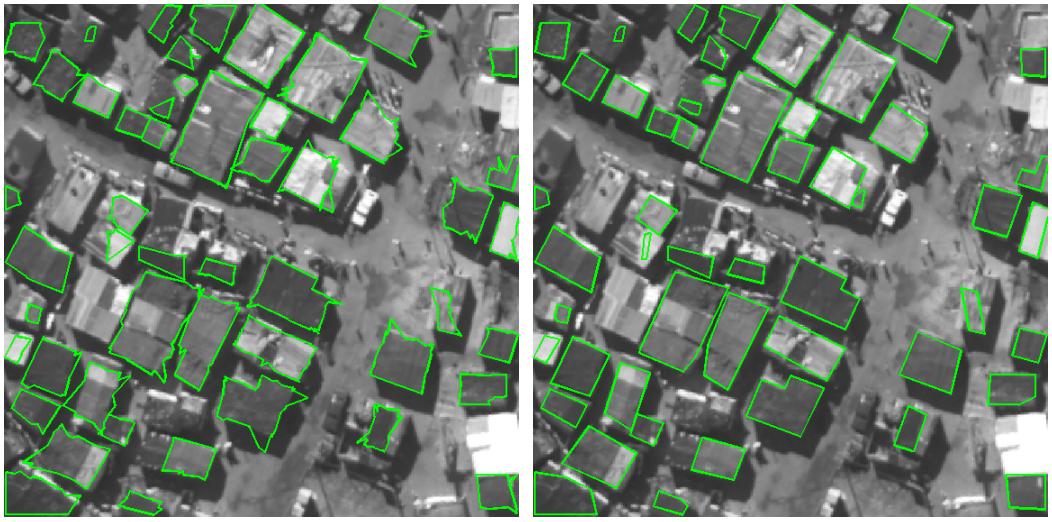


Figure 5.41: Comparing intersection point support for different edge configurations. Edges are drawn as thick solid lines; edge gaps are drawn as dashed lines; the intersection point for which the support is being calculated is marked with an "X". Tick marks along the edges and edges gaps indicate lengths of 1 unit. Point support decreases from A to E.

Point support for differing edge configurations is given in Figure 5.41. The different edge configurations in this figure are all drawn to the same scale. The point in A has the strongest support, and the point support decreases from A through E, although the intersection points in C and D have identical support. Comparing A and C, it is natural to expect A to have larger support because it has complete edge line coverage, whereas C has a large edge gap on one side. B has a higher support value than C because, even though the total length of the supporting edges is identical, B has a smaller edge gap. C and D have identical support because although C has one long edge this is negated by the short, distant second edge participating in the intersection. D, on the other hand, has two short, but well localised, edges with respect to the intersection. Finally, E, has the lowest point support because the supporting edges are short and there are large edge gaps.

Applying model-driven simplification to the expanded boundaries using Algorithm 5.1 with an objective function expressed by Equation 5.20 results in the boundaries depicted in Figure 5.42b. The boundaries generally delineate the shacks with greater accuracy than those prior to expansion (Figure 5.40a).



(a) Expanded reference boundaries.

(b) Expanded reference boundaries after model-driven simplification.

Figure 5.42: Model-driven simplification of expanded reference boundaries for the Marconi Beam image.

The approach here differs largely from [135] and perceptual grouping in general, in that strong use of an existing reference boundary is made and edges are only used indirectly in generating corner and other boundary points.

Constraints at a local and more global level play an important role. Local constraints are used to filter out edges which are not aligned with individual reference boundary segments. Global shape constraints are applied during model-driven simplification of the expanded boundary wherein rectilinearity, compactness and point support are maximised over the entire boundary, and the boundary's canonical orientation is anchored to a certain degree.

5.8.4 Three Phase Boundary Regularisation and Localisation

It is worthwhile reviewing the different phases that are involved in production of the final hypothesis boundary and how they are related to one another. In all, there are three distinct phases:

1. Removal of digitisation noise using DCE.
2. Model-driven simplification based on global shape measures – rectilinearity and compactness.
3. Boundary expansion through edge endpoints and intersection points, followed

by model-driven simplification based on global shape measures and image evidence (point support).

The first phase involves the removal of fine scale detail along the boundary which mostly corresponds to digitisation noise. During this pre-processing phase boundary regularisation and localisation are not dealt with; the concern is in trying to establish the "true" shape that the boundary represents. This allows shape measures to be reliably calculated and these measures are key aspects of the follow-on phases. The second phase is one of regularisation. No image evidence is used in further abstracting the boundary to a form which fits the assumed model — a 4–6 sided compact and rectilinear shape. The third and final phase involves components of localisation, as boundaries are expanded using points determined from image edges, and regularisation, as boundaries are re-simplified to conform to the model.

Figure 5.43 is a summary figure, grouping the results of the different phases together for selected hypotheses from the Marconi Beam image. In each row, moving from left to right, it is possible to see the progression from the original boundary to the final boundary. The localisation of the boundaries improve, in some cases quite significantly. A quantitative analysis of this improvement is given in Chapter 6 for the Marconi Beam image and others.

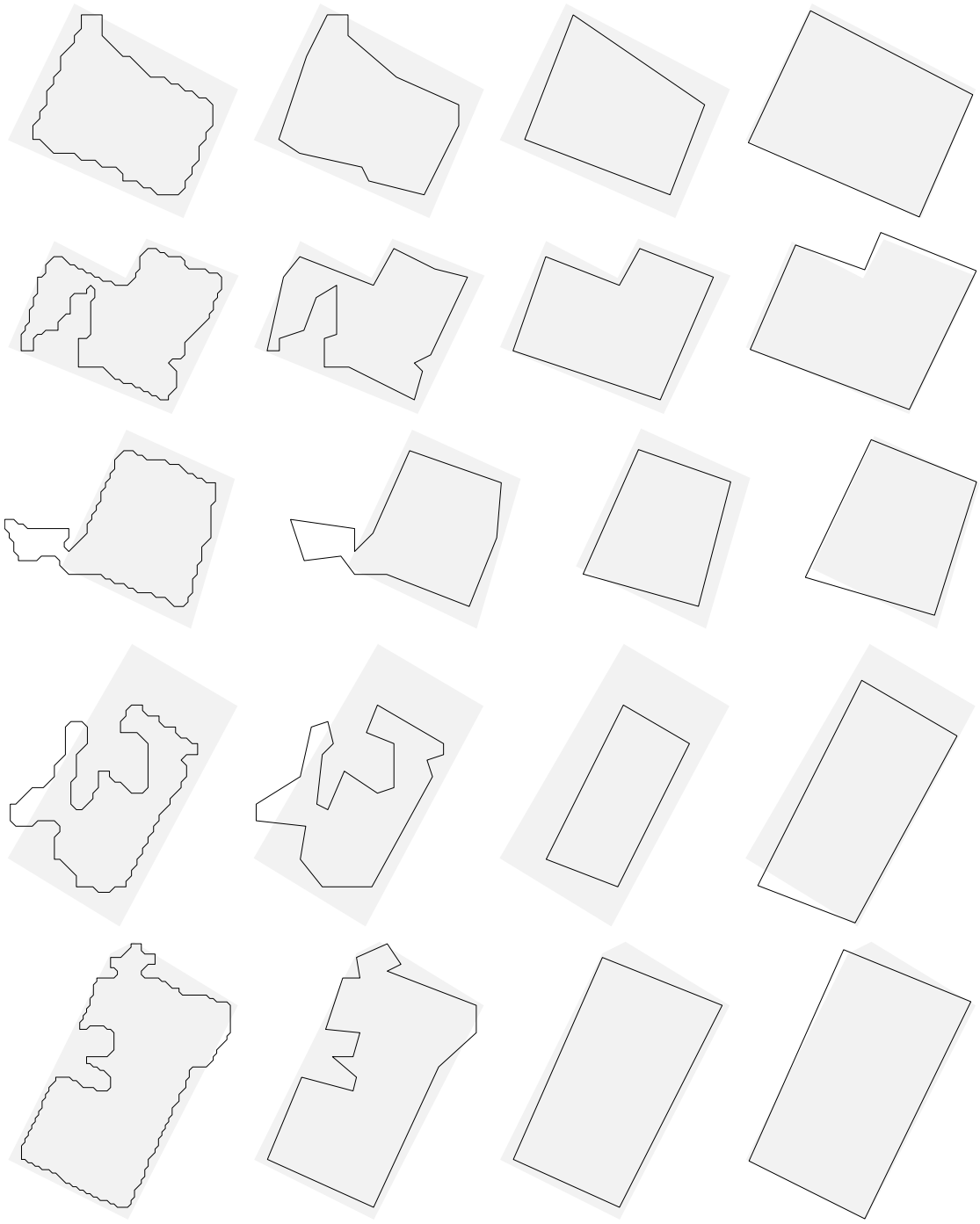


Figure 5.43: Three phase regularisation and localisation of hypotheses' boundaries. Left Column: Original region boundaries; Middle-Left Column: boundary after removal of digitisation noise; Middle-Right Column: boundary after model-driven simplification (regularisation); Right Column: boundary after expansion and re-simplification (localisation and regularisation). In each diagram, the shaded area represents the ground truth for the shack in question.

5.9 Conclusion

This chapter has presented an approach to shack detection using a single, nadir-view source image of an informal settlement. Key to the detection strategy is the construction of an anisotropic scale-space with homogeneous regions, which are viewed as shack hypotheses, being extracted and verified at each scale. The strategy is region-based and, therefore, issues related to boundary digitisation noise, localisation and regularisation feature prominently. The shape measures of rectilinearity and compactness are used both for the regularisation of hypotheses' boundaries so that they conform to the shack model, and as an input to a fuzzy inference system which non-linearly combines the evidence for, and ranks overlapping, and competing, hypotheses. Using the fuzzy system allows for the creation of rules which reflect knowledge about the value of evidence within the context of the overall performance of the system. Grouping of hypotheses plays a small but important role in allowing non-shadow-verified hypotheses, which may correspond to shack roof fragments, to form part of the final scene interpretation.

The next chapter presents the results achieved when applying the detection strategy described here to a number of test images. Detailed qualitative and quantitative results are given and the performance of this system is compared to existing systems.

Chapter 6

Results

A qualitative and quantitative assessment of system performance is conducted. The different quantitative metrics which have been used for evaluating performance are presented and explained. Detailed results for the 'Marconi Beam 1' image are given along with a summary of the results for all of the other images tested. These results are compared to those from other shack detection systems and found to be favourable.

6.1 Introduction

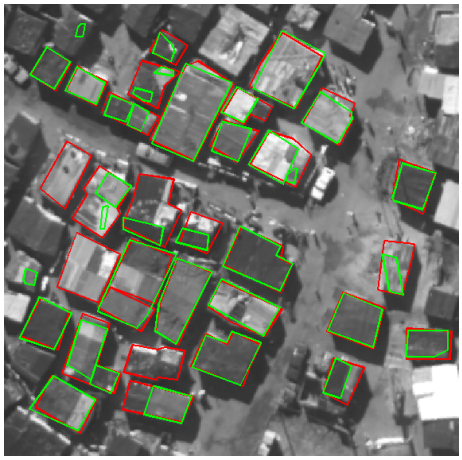
In this chapter the results of the system for different images are presented and analysed. Detailed results are presented only for the 'Marconi Beam 1' image, while, the results for all of the other images tested are given in a summarised format. The detailed results for these additional images can be found in Appendix A.

For the 'Marconi Beam 1' image performance is both qualitatively and quantitatively assessed. Qualitative assessment is enabled by the production of images which allow the extracted hypotheses to be visually compared with the ground truth. Quantitative assessment is performed through calculating standard performance metrics. The system's performance at different detection stages, and for different hypotheses' boundaries is also considered. A brief evaluation of the execution time for each processing step is given.

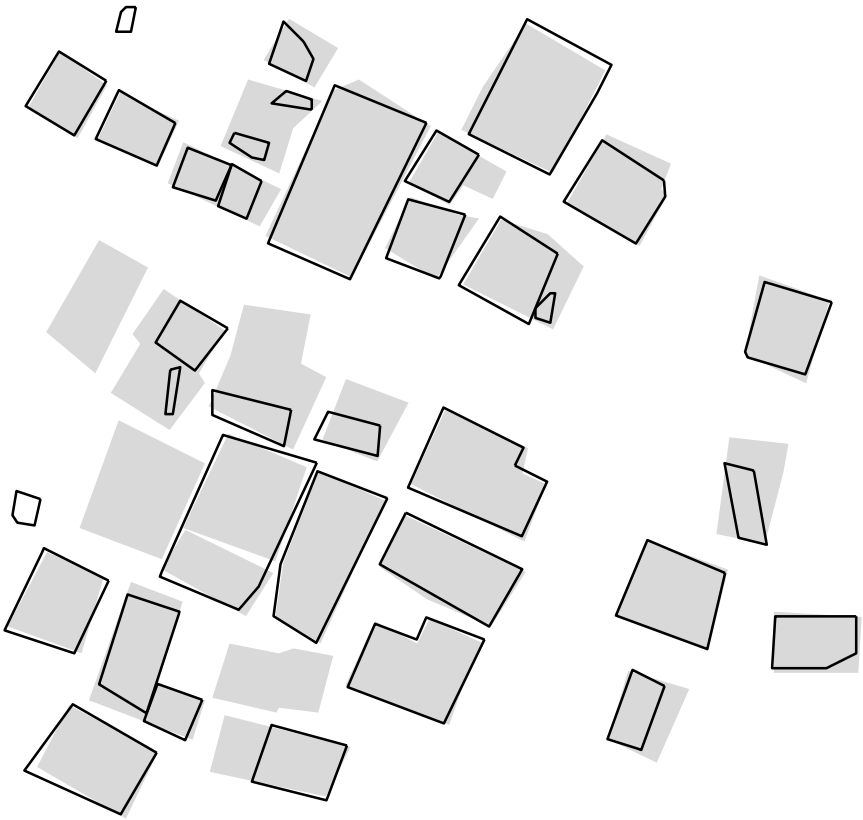
The chapter concludes with a discussion of the results from all of the test images which leads to an understanding of the system's strengths and weaknesses. Furthermore, the results are compared to those from other shack detection/extraction systems.

6.2 Qualitative Assessment of Performance

(a) Source image.



(b) Ground truth polygons (in red) and extracted hypotheses (in green) overlaid on the source image.



(c) Schematic of ground truth (shaded areas) overlaid with extracted hypotheses.

Figure 6.1: Results for the 'Marconi Beam 1' image.

Figure 6.1a presents the 'Marconi Beam 1' source image, while, Figure 6.1b presents the source image with the ground truth and extracted hypotheses overlaid. To allow for an easier visual comparison, a schematic of the ground truth and extracted hypotheses is given in Figure 6.1c. These figures are presented for each of the additional images tested in Appendix A.

When analysing performance, special care needs to be taken to deal with edge effects. The detection system has been designed to detect structures that appear completely in the image along with any shadows cast. If either a shack/building or its shadow is severely truncated by an image border then it is unreasonable to expect the system to identify it correctly. One way of dealing with this is to create "buffer zone" around the border of the image. All ground truth and system hypotheses which lie within a certain distance of the image border are excluded when calculating performance metrics. This works well if the image is rotated so that roads surrounding blocks of shacks/buildings predominantly align with the image borders, and none or very few shacks lie within the buffer zone. All of the additional images tested (Appendix A) have been rotated so that this is the case. The 'Marconi Beam 1' image has not been rotated and, therefore, there are instances where system hypotheses have been included in the analysis but the corresponding ground truth, which lies partly within the buffer zone, has been excluded (see the two false positives in Figure 6.1c).

It is evident from Figure 6.1c that model-driven simplification of hypothesis boundaries forces the majority of boundaries to assume a fairly rectilinear shape. Most of the shacks are well detected; however, in some cases, shack roofs are extracted as one or more fragments or are missed entirely.

The use of shadow is key to hypothesis verification, so if there is insufficient shadow along a particular roof-shadow boundary, either because the shadow is mostly occluded by a neighbouring shack or because it is not detected because the shadow threshold is too low, then the shack in question will not be detected.

Even when there is sufficient shadow, a shack may appear as undetected or only partially detected in the final interpretation of the scene. The interplay between the contrast among different roof materials, the contrast between the borders of a shack roof and its surroundings, and the choice of the flux function peak (Figure 5.4b) can result in the formation of homogeneous regions over different scales which either never fully encompass a particular shack's roof or fail to be entirely contained within the roof, or both (Figure 5.7). The resultant boundaries will be poorly aligned with the ground truth at points and none may be verified, in which case the shack

will be missed entirely. If one or more of these boundaries do succeed in being verified, subsequent processing steps could still fail to enlarge the selected boundary, or to correct its deformations due to flooding, or to group it with other boundaries delineating fragments of the same roof. In all of these cases the correspondence between the system's hypothesis and the ground truth will be weaker, and the shack will be partially detected.

Finally, it is worth noting, that the detection strategy relies on the scale-space being sufficiently well-sampled. If this is not the case then the detection performance will be adversely affected.

The false positives in Figure 6.1c are due to edge effects. Genuine false positives are evident in some of the other test images (see Figure A.1c for example). False positives occur when an extracted homogeneous region is found to have sufficient shadow support (the homogeneous region occurs in a position capable of casting the detected shadow) but the region does not, in fact, correspond to a shack roof. False positive hypotheses may have sufficient shadow support for the following reasons:

- The hypothesis corresponds to some non-shack object which is casting shadow. For instance, significant shadow can be cast by walls (which may be part of a shack under construction), vegetation and cars.
- The hypothesis delineates some area of the image which is adjacent to a dark, non-shadow region of the image which has been misclassified as shadow. Misclassification occurs when areas of the image which are not shadow have intensities below the shadow threshold.

6.3 Quantitative Assessment of Performance

Various metrics (see Section 2.8 for the background discussion) are used in assessing the overall performance of the system on each image as well as the accuracy with which individual shacks are detected. These are presented in the following subsections.

6.3.1 Detection and Quality Metrics

Established metrics (Equation 2.1) are used in assessing the system's performance for each image. These are the branching factor, miss factor, detection percentage, and quality percentage:

$$\begin{aligned} \text{Branching Factor} &= \frac{\text{FP}}{\text{TP}}, \\ \text{Miss Factor} &= \frac{\text{FN}}{\text{TP}}, \\ \text{Detection Percentage} &= 100 \cdot \frac{\text{TP}}{\text{TP} + \text{FN}}, \\ \text{Quality Percentage} &= 100 \cdot \frac{\text{TP}}{\text{TP} + \text{FP} + \text{FN}}. \end{aligned} \tag{2.1}$$

where

FP = false positives
TP = true positives
FN = false negatives

These metrics are calculated based on both building counts and area. For building-count metrics, an extracted hypothesis is regarded as a true positive if any part of it overlaps with a ground truth hypothesis; otherwise it is deemed to be a false positive. Ground truth hypotheses which are not overlapped by any extracted hypotheses are counted as false negatives. Area-based metrics are determined using the true and false positive, and the false negative polygonal area for the entire image. Note, that if two extracted hypotheses overlap each other (as in Figure A.1c), the overlap area is not counted twice.

Building-Count Metrics		Area-Based Metrics	
Buildings Detected (TP)	29	Branching Factor	0.06
Buildings Missed (FN)	3	Miss Factor	0.44
Non-buildings Detected (FP)	2	Detection Percentage	69.58
Detection Percentage	90.63	Quality Percentage	66.71
Quality Percentage	85.29		

Table 6.1: Performance metrics for the 'Marconi Beam 1' image.

The metrics for the 'Marconi Beam 1' image are given in Table 6.1. Over 90% of the shacks are detected and almost 70% of the shack roof area is extracted. These metrics correspond with what appears to be a fairly good performance qualitatively. The area-based detection percentage calculates the ratio of extracted true positive

area to the ground truth area (TP + FN). This is a stricter measure than using building counts and, consequently, the detection percentage is around 20% lower. The miss factor illustrates that for every unit of shack area detected, just under half a unit is missed.

Quality metrics are more rigourous than detection metrics, taking into account the false positives that are detected. For this image, the quality percentages are close to the detection percentages, indicating that there are few false positives and the false positive area in total is small. This is also evident when examining the branching factor, which is low at 0.06. In other words, for every unit of area found that is a shack, 0.06 of a unit is found that corresponds to the background.

6.3.2 Metrics per Stage

One of the side effects of using a region-based detection strategy is that closed boundaries can be derived for hypotheses from almost the start of the detection process (unlike in edge-based approaches). This enables the calculation of performance metrics at multiple stages, allowing for an analysis of the effect of each stage.

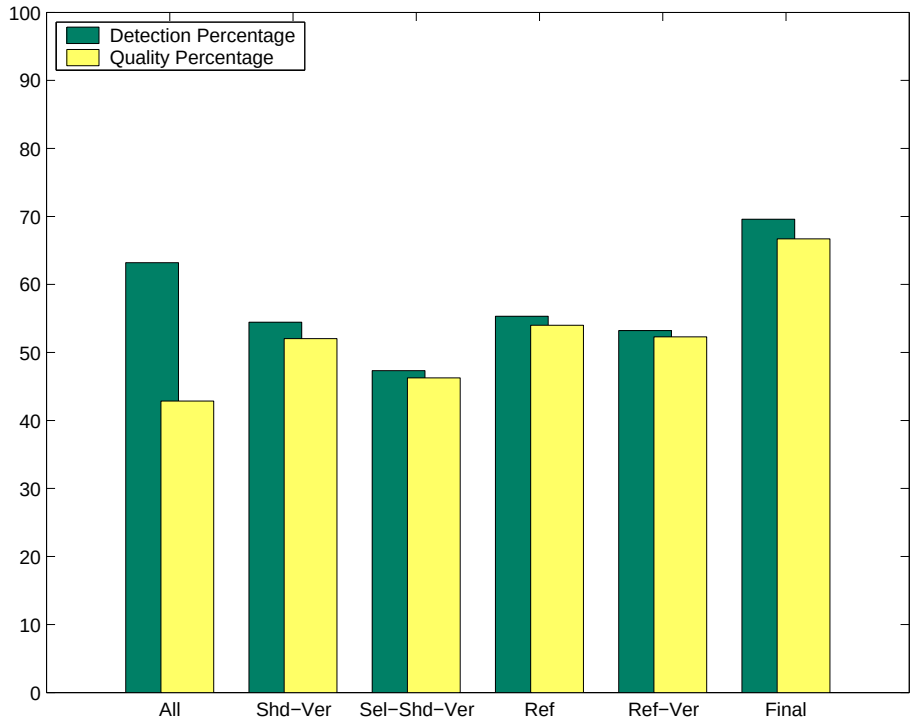


Figure 6.2: Area-based performance metrics for the 'Marconi Beam 1' image at different stages of the detection process. Boundaries resulting from model-driven simplification are used wherever possible.

Figure 6.2 presents the area-based detection and quality metrics at a number of different stages during the interpretation of the 'Marconi Beam 1' image. The ground truth area used in calculating the metrics at each stage remains identical, though the set of extracted hypotheses and the area that they cover varies. The set of extracted hypotheses, at each of the stages considered, is taken to be as follows:

- *All* – the noise-free boundaries of extracted regions which are within the expected size range and do not overlap shadow, as shown in Figure 5.16b.
- *Shd-Ver* – the shadow-verified model boundaries as shown in Figure 5.22b.
- *Sel-Shd-Ver* – the selected hypotheses from conflicting sets, as in Figure 5.28b.
- *Ref* – the reference boundaries prior to expansion but post-grouping, as in Figure 5.40a.
- *Ref-Ver* – the verified reference boundaries (based on edge shadow-support).
- *Final* – the expanded and re-simplified (re-regularised) reference boundaries, that is, the final boundaries as in Figure 6.1b.

In all cases where extracted hypotheses overlap, the union of these hypotheses is used. Detection and quality metrics have been calculated using boundaries which have been simplified according to the model, where possible, in order to remove the effect of the actual boundary type on the metrics (see Section 6.3.3 for an analysis based on boundary type). Note, however, that for the first stage, *All*, noise-free boundaries are used as the model-driven-simplified boundaries have not yet been derived at this point in the detection process.

From the graph, it is evident that the initial stage has a relatively high detection rate coupled with a low quality percentage. This is understandable because, at this stage, very few hypotheses have been eliminated so the likelihood of ground truth area being covered is high. On the flip side, the quality is poor as many of the hypotheses overlap background area in the image.

After shadow verification (*Shd-Ver*) the quality of the interpretation is substantially increased as many false positives are removed. It also decreases the detection rate. This is mostly due to fragmented roof detections. Fragments which happen to be positioned away from the shack's shadow are removed in the verification step. Additionally, in some circumstances, shadow support is poor and true positives are eliminated.

Resolving conflicting shadow-verified hypotheses using the fuzzy inference system produces the results annotated *Sel-Shd-Ver*. Here, both the detection and quality

percentages are reduced compared to the previous stage. This can be explained as follows.

In Figure 5.22b the majority of conflicting hypotheses designate true positive area but the hypotheses are not necessarily contained within one another. In such cases, selecting a single hypothesis from the set reduces the overall detection percentage as the union of overlapping hypotheses covers a greater proportion of the ground truth than any single hypothesis in the overlapping set. By inspecting Figures 5.28a and 5.28b it is also clear that the hypothesis which is selected from a conflicting set is not always the best choice, that is, the hypothesis which maximises the quality percentage for the shack concerned. In a small number of cases, hypotheses exist which are substantially incorrect in that they overlap a large amount of non-shack area. These incorrect hypotheses are not selected by the fuzzy rules, implying an increase in the quality percentage in these particular cases. However, these cases are in the minority so the net effect is a decrease in both measures.

The detection and quality percentages at the *Sel-Shd-Ver* stage are very close to one another. This indicates that the extracted area at this point is of high quality even though the detection rate is comparatively less than that of other stages. The hypotheses at this stage, therefore, form a reliable foundation for expansion.

At the reference boundary stage (*Ref*), non-shadow-verified hypotheses having sufficient combined support have been identified and grouping of hypotheses has taken place. There is an improvement in the metrics as a small number of non-shadow-verified hypotheses in the 'Marconi Beam 1' have been identified and a few groupings have occurred. This improvement demonstrates the utility of considering non-shadow supported hypotheses and of grouping, even though the number of groupings is small.

The reference boundaries are verified according to the presence of at least one edge with strong shadow support which is aligned with the roof-shadow boundary. The metrics calculated for the verified boundaries are annotated *Ref-Ver* on the graph. This verification process is aimed at reducing the number of false positives. However, verification typically involves a tradeoff between the true positive area and the false positive area. This tradeoff occurs because the verification criterion is unable to cleanly separate true positives from false positives. In terms of the performance metrics, verification tends to narrow the gap between the detection and quality percentages (as false positives are eliminated) while simultaneously decreasing both (as some true positives are also eliminated).

Metrics calculated from the final interpretation of the scene (labelled *Final* in Figure 6.2) show a substantial improvement over those for the verified reference boundaries. For the 'Marconi Beam 1' image the detection and quality percentages increase by more than 10%. This is due to the fact that reference boundaries sit within the borders of each shack roof while the final boundaries are far better localised. Comparing the metrics for the last two stages validates this quantitatively. The metrics per stage vary in a similar manner for all of the images tested (Appendix A).

6.3.3 Metrics per Boundary Type

An important part of the detection process is boundary regularisation and localisation. The particular methods used are discussed in various sections of the methodology chapter and Section 5.8.4 reviews the three distinct phases that are involved. It is interesting to analyse and understand exactly how the different boundary types affect performance. In order to perform this analysis, hypotheses belonging to the final interpretation of the scene are flagged. The area-based detection and quality metrics are then calculated for only these hypotheses while varying the boundary type. The result for the 'Marconi Beam 1' image is the graph given in Figure 6.3. The metrics are grouped per boundary type with the types identified as follows:

- *Original* – the original 8-connected boundaries derived from the extracted homogeneous regions.
- *Noise-free* – the original boundaries after the removal of digitisation noise (by applying DCE).
- *Model-Driven-Simplified (MDS)* – the noise-free boundaries after model-driven simplification (Algorithm 5.1).
- *Expanded* – the model-driven simplified boundaries after expansion through the use of image edges.
- *Re-Simplified* – the expanded boundaries after model-driven re-simplification (Algorithm 5.1 using Equation 5.19).

The metrics for the noise-free boundaries approximate those of the original boundaries. This is to be expected as the transformation from the one boundary type to the other is minor. This is also desired as noise removal should "reveal" the true shape of the original boundary but not introduce strong deformations.

It is interesting to see that model-driven simplification of the noise-free boundaries produces no significant improvement in the metrics for this image. In fact, there is a

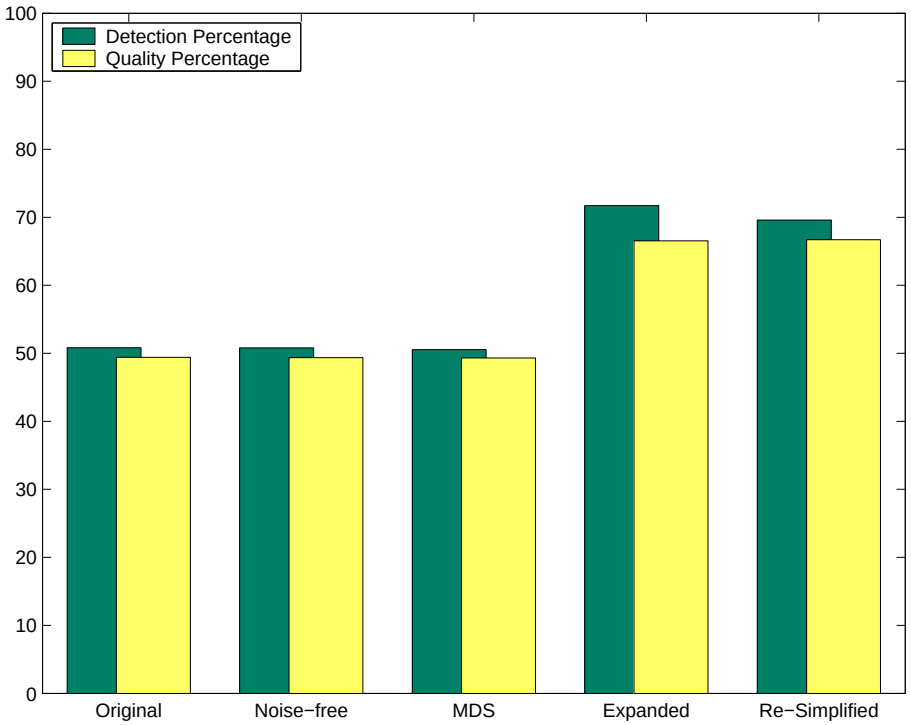


Figure 6.3: Area-based performance metrics for the hypotheses in the final interpretation of the 'Marconi Beam 1' image with differing types of boundaries.

slight decrease in performance though this is not the case for all of the images tested. This implies that even though strong deformations in the noise-free boundaries are removed (see Figure 5.43) the net true positive area, on the whole, remains about the same. This finding does not negate the value of applying model-driven simplification to the noise-free boundaries because the regularised boundaries generally exhibit greater fidelity with the actual shack/building boundaries. The model-driven simplified boundary becomes the reference whereby aligned edges are found for boundary expansion. If the noise-free boundary was used, deformations would cause spurious edges to be included in the expansion process. Additionally, grouping which relies primarily on the rectilinearity would not be viable with noise-free boundaries.

The expanded boundaries belonging to the hypotheses in the final interpretation of the image are shown in Figure 5.40b. As shown in Section 5.8.2 the expanded boundaries sit outside of the reference (MDS) boundaries and, hence, cover a greater amount of true positive area. The false negative area remains unchanged as no new hypotheses are generated. These factors cause the detection rate to increase substantially. Expansion may also introduce significant deformations from the true boundaries at certain points (often near corners) and these deformations increase the false positive area. On balance, the increase in true positive area is far greater than the increase in false positive area, with the result that the quality percentage

also rises dramatically.

Finally, the expanded boundaries are re-simplified which causes a slight reduction in the performance metrics but produces nicely regularised boundaries. Taken together, boundary expansion and re-simplification provide a significant performance gain as they improve the localisation of the hypotheses boundaries.

The pattern of variation exhibited in Figure 6.3 for metrics per boundary type is echoed in the other images tested (Appendix A). The only point of difference is that the MDS boundaries may offer an improvement over noise-free boundaries in some cases.

6.3.4 Shape Accuracy Metrics

The above metrics are concerned with the system's overall performance on an image. Performance can also be evaluated on a per-shack/building basis. This form of analysis gives insight into the accuracy with which individual shacks are being detected.

Three shape accuracy measures are calculated, namely the DP shape accuracy, the QP shape accuracy and the shape accuracy measure used in [61] (based on a measure in [63]).

In order to calculate these measures, it is necessary to associate the system hypotheses with the ground truth. An association between an hypothesis polygon and a ground truth polygon is formed if there is overlap between the two. A hypothesis polygon may be associated with (that is, overlap) more than one ground truth polygon and multiple hypothesis polygons may be associated with a single ground truth polygon. The associations between hypotheses and ground truth polygons are used in calculating the shape accuracy. It is important to note that errors of commission (false positives) and omission (false negatives) are not included in this analysis — shape accuracy is only calculated in cases where there is some correspondence between a ground truth polygon and a hypothesis polygon.

The DP and QP shape accuracy measures are simply the area-based detection and quality percentages (Equation 2.1) determined on a *per-shack* basis. The particular polygons involved in the calculation for each shack are the shack's ground truth polygon and any hypothesis polygons associated with it (as described above). The detection percentage determines the amount of shack roof area that is found for

a particular shack. The quality percentage is a stricter measure of accuracy and decreases the detection percentage according to the amount of background area that is found along with the shack roof area.

Rüther et al. [61] use the following shape accuracy measure:

$$\text{Shape Accuracy} = (1 - (|A - B|)/A) \tag{6.1}$$

where

A = area of a shack/building in the ground truth

B = area of the corresponding extracted shack/building

This measure not only expresses the completeness with which shacks are extracted but also penalises cases where the extracted area is larger than the ground truth area ($B > A$). However, this measure is not that reliable because, for instance, A and B might have almost identical areas but only partially overlap. The shape accuracy, as defined above, will produce a result close to 1 or 100% even though the actual roof area extracted is small. In assessing the shape accuracy quality, the QP measure is preferred; however, the shape accuracy as in Equation 6.1 is calculated in order to enable a direct comparison with previous research.

Other similar shape accuracy measures are presented in the literature, for example, the *area coverage rate* in [12], defined as a building's ground truth area divided by the corresponding extracted area.

Shape Accuracy Metrics (%)	
DP Mean Accuracy	75.19
QP Mean Accuracy	68.98
Mean Accuracy (Equation 6.1)	67.69

Table 6.2: Shape accuracy metrics for the 'Marconi Beam 1' image.

Table 6.2 summarises the shape accuracy metrics for the 'Marconi Beam 1' image. The mean accuracy for each measure is given. For example, the DP mean can be interpreted as follows: on average when a shack roof is detected, just over 75% of the roof area is found. The QP mean and the measure expressed by Equation 6.1 indicate the average quality of extraction. Values closer to the DP mean imply a better quality of extraction, that is, less background (or other shacks) is/are found when extracting a single shack.

6.4 Stack Level Contributions to Final Interpretation

The motivation for adopting a scale-space approach to shack detection is that many shacks appear more clearly in abstracted versions of the source image, due to the annihilation of roof substructure. Altering the appearance of shacks in such a manner, facilitates their detection at scales other than the original.

In order to validate the above assertion, it is necessary to determine from which stack levels the final hypotheses are derived. Figure 6.4 depicts a histogram showing the frequency with which the nine different stack levels give rise to the hypotheses appearing in the final interpretation of the scene. Recall, that hypotheses for a given shack are generated at multiple scales. One of these competing hypotheses is judged by the fuzzy rule system as most likely to correspond to the true shack outline. The stack level that contains this hypothesis is understood to have given rise to it in the final interpretation.

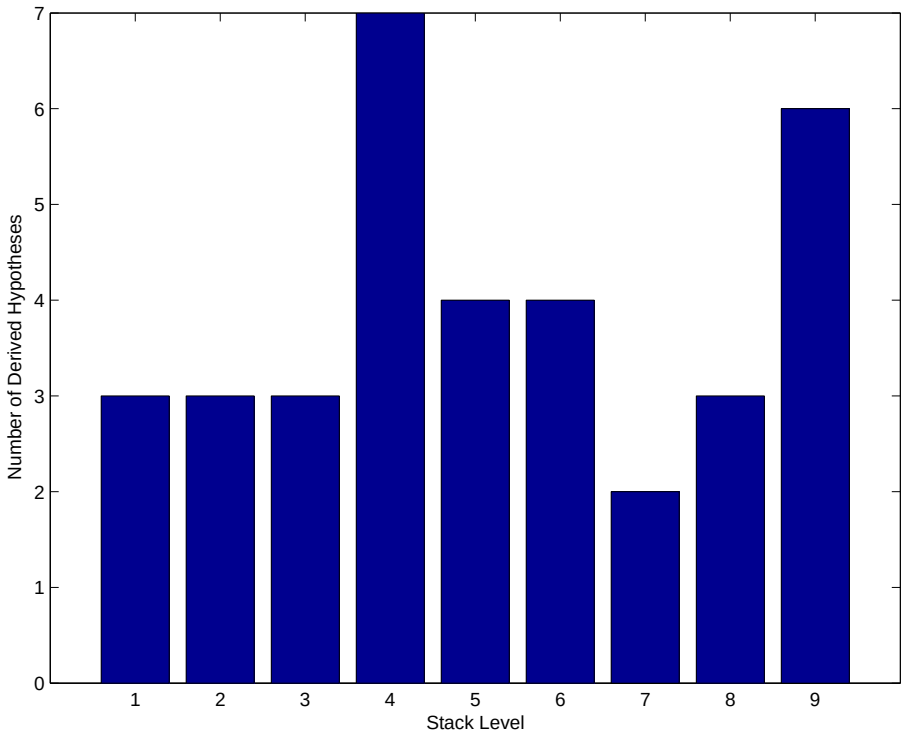


Figure 6.4: Histogram of stack levels from which final hypotheses are derived for the 'Marconi Beam 1' image.

From the histogram it is clear that the final hypotheses come from many different scales or stack levels. It is noteworthy that the original, non-diffused, image only gives rise to a small percentage of the total number of hypotheses. Additionally,

there appears to be no single best scale/stack level at which to analyse the image as each level contributes hypotheses to the final result.

6.5 Execution Time Evaluation

The detection system described has been implemented using MATLAB (version 6.5, release 13), a product of the MathWorks company [137]. MATLAB is a language for technical and scientific computing. It also offers strong visualisation capabilities and includes image processing, fuzzy logic and other toolboxes. The language is more high-level than, for example, C, with the consequence that it affords rapid prototyping but at the expense of execution speed. MATLAB is a good choice for prototyping and demonstrating the viability of the detection strategy, but not necessarily for a commercial implementation, where tight GIS integration will probably be important.

No real attempt has been made at optimising the prototype in terms of speed. Nevertheless, it is helpful to briefly consider the performance of the system, in terms of execution time, on the 'Marconi Beam 1' image in order to understand the bottlenecks, and to explore avenues for improvement. This image, along with the others, has been processed on a single core, Pentium 4 desktop personal computer running at 3 GHz with 1 GB of RAM.

The timings in Table 6.3, although recorded for single run on the 'Marconi Beam 1' image, are indicative of the relative run-time performance of each stage. The timings of stages which involve non-shadow-verified hypotheses (the right-hand branch of Figure 5.30) have not been included. These stages repeat the same kinds of computation as on the left-hand branch but on different sets of hypotheses. Therefore, the stages listed in Table 6.3 involve the entire range of computations utilised in the system, and are sufficient for gauging the relative run-time performance of each.

The first column of Table 6.3 describes the stage. The second column gives the number of hypotheses that are processed in the stage, that is, the number of input hypotheses. The third column gives the number of output hypotheses. This will remain unchanged from the input if the stage does not involve filtering of hypotheses. The fourth column presents the run-time, in seconds, of the stage while the fifth column provides the number of seconds taken per hypothesis (where it makes sense). The final column gives percentage time that each stage takes of the total time (31.27 minutes).

Stage(s)	Hypotheses In	Hypotheses Out	Run-time (seconds)	Seconds/ Hypothesis	Percentage of total time
Scale-space construction + region extraction	-	612	124.8		6.61
Linking regions across scales	612		42.5		2.27
DCE of all hypotheses	612		238.8	0.39	12.72
Shadow support for all hypotheses (NF)	612	294	720.6	1.18	38.38
Model-driven simplification of shadow-verified hypotheses	294		118.8	0.40	6.33
Shadow support for shadow-verified hypotheses (MDS)	294		264.0	0.90	14.07
Shadow-verified hypothesis selection over scale	265	65	9.08		0.48
Grouping and verifying reference boundaries	72	59	84.04	1.17	84.4
Boundary expansion	59		23.5	0.40	4.50
Boundary re-simplification using model	59		250	4.23	13.33

Table 6.3: Evaluation of run-time performance. NF = Noise-free boundary; MDS = Model-driven simplified boundary.

It is apparent from the figures in Table 6.3 that some of the earlier stages dominate in terms of the percentage of total time taken, particularly, the calculation of shadow support for all hypotheses. This is understandable as in the early stages of analysis the number of hypotheses to be processed is very large — hypotheses are generated from each scale and many represent false positives. As analysis progresses, more and more hypotheses are filtered out until eventually the final set of hypotheses is arrived at. In this case, as is typical, the number of final hypotheses is less than 10% of the original number.

The relative performance per-stage can be ascertained from the seconds-per-hypothesis column. The more time-costly stages are considered below.

The calculation of shadow support, be it for model boundaries or simplified boundaries, is fairly expensive as the image is sampled at tens of points. Calculating shadow support for regularised boundaries is quicker than for noise-free boundaries, because processing a fewer-sided (4–6 sides) regularised boundary requires less processing than the many-sided noise-free boundary from which it originates.

The grouping and verification stages also takes a relatively long time. These stages include a number of distinct steps:

1. grouping of boundaries,
2. edge detection, and
3. verification of reference boundaries.

Of these, verification takes more than half the time as shadow-support score is calculated for each and every edge aligned with the roof-shadow boundary.

Finally, model-driven simplification of expanded boundaries is also expensive due to the large increase in the number of boundary segments per hypothesis and consequent cost in determining rectilinearity, and the fact that point support now forms part of the regularisation process.

The total time taken to process the ‘Marconi Beam 1’ image is just over 30 minutes, which is longer than a human photo-interpreter would take. However, the system’s run-time performance could be improved dramatically, possibly by more than an order of magnitude, as it is very amenable to parallel processing on multi-core processors or distributed processors. After the creation of the image stack, each image (stack level) can be processed individually. Regions can be extracted, hypotheses’ boundaries simplified, support calculated and so on. Furthermore, each individual hypothesis in

an image can be processed independently of others in the same image. The results only need to be integrated at points where conflicts need to be resolved across stack levels, and when deriving the support for hypotheses from other hypotheses. The final few stages of the detection process, post conflict resolution, cannot be parallelised on an image-wide basis but, at this point, the total number of hypotheses has been substantially reduced.

As the photo-interpretation task scales in size (such as when large photo-mosaics or time-series images of a particular area need to be processed), so does the benefit to be gained by automating part of the work.

6.6 Results Summary and Discussion

Appendix A presents the detailed results for all of the test images. The performance metrics are summarised in Tables 6.4 and 6.5.

	TP	FN	FP	Building Count DP	Building Count QP	Area DP	Area QP
Marconi Beam 1	29	3	2	90.63	85.29	69.58	66.71
Marconi Beam 2	72	5	14	93.51	79.12	70.35	63.44
Imizamo Yethu	20	4	12	83.33	55.56	54.64	46.33
Sparse Rural	14	0	1	100.00	93.33	83.79	77.17

Table 6.4: Summary of detection and quality percentages.

	DP Mean Accuracy	QP Mean Accuracy	Mean Accuracy (Equation 6.1)
Marconi Beam 1	75.19	68.98	67.69
Marconi Beam 2	74.43	64.64	60.81
Imizamo Yethu	59.61	54.49	59.66
Sparse Rural	82.61	75.95	79.38

Table 6.5: Summary of shape accuracy metrics.

In Table 6.4 'Marconi Beam 1' refers to the image used in explaining the detection strategy. 'Marconi Beam 2' is another image section taken from the Marconi Beam dataset [83, 5]. The ground pixel resolution of this dataset is 0.18 m. 'Imizamo

Yethu' depicts a portion of the Imizamo Yethu settlement near Hout Bay which is close to Cape Town, South Africa. The image forms part of a dataset produced in 1997 for a collaborative research project funded by AusAid's Australia-South Africa Institutional Links Program [138]. The resolution is similar to that of the Marconi Beam dataset.

These images represent typical South African informal settlements, in that shacks are mostly single storied with flat roofs which are constructed from diverse materials. Their geometry is simple (mostly 4-sided), and they are quite densely built. Additionally, their surroundings are characterised by a general lack of vegetation. The 'Imizamo Yethu' image, however, does contain a number of trees.

The final image, 'Sparse Rural', depicts a rural South African scene in which the buildings are more characteristic of a formal settlement. The building roofs have A-frame shapes and the roof material used is uniform. This image has been included in order to establish continuity with previous work done by the author [115] and to demonstrate the capability of the system on images which differ in some ways from informal settlements. This image is unrectified and has been produced by scanning an ordinary aerial photograph.

Performance on all images in terms of building counts is good with percentages upwards of 83% being achieved. This is born out by the fact that the number of false negatives in each case is relatively small. Building count quality percentages are also high with the exception of 'Imizamo Yethu'. The area-based detection percentage is 69.5% and upwards for all images (except 'Imizamo Yethu') with the quality percentage being around 5% less on average. Having quality percentages which are close to the detection percentages implies that the number of false positives or false positive area is small relative to true positives and false negatives.

The performance on 'Imizamo Yethu' stands out as being particularly poor. This can be attributed to the presence of large trees in the scene which overshadow and obscure a number of shacks and poorer overall image contrast, as compared to the other images.

The system performs best on 'Sparse Rural'. This is because:

- The buildings are well separated, allowing for unoccluded shadows.
- The roof material is uniform which removes most of the difficulties of dealing with roof substructure.

- There is minimal vegetation.
- The contrast of the image is good.

The 'Sparse Rural' scene is quite different from the informal settlement scenes and this manifests itself in differences in the detection stages. In Appendix A:Figure A.9a it is apparent that there is a comparatively large difference between the detection and quality percentages for the first stage ('All'), at which the hypothesis set consists of all extracted region boundaries. This is because the scene depicts a large amount of unpopulated terrain and many of the extracted homogeneous regions correspond to this. In other scenes the unpopulated area is much smaller.

Another difference, visible in Figure A.9a, is that the improvement in performance from the 'Sel-Shd-Ver' stage to the 'Ref' stage is greater than for other images. This is because all of the building roofs in 'Sparse Rural' have an A-frame shape and the difference in contrast between the sun-facing roof panels and the non-sun-facing panels is large. Consequently, each panel is extracted as a separate region. At the point in the detection process where hypotheses are verified by shadow, most of the hypotheses corresponding to sun-facing panels are excluded from the verified set. Shadow-verified hypotheses are then grouped with non-shadow-verified hypotheses to form reference boundaries in the 'Ref' stage. At this stage, the sun-facing roof panel hypotheses have been grouped, mostly successfully, with the shadow verified (non-sun-facing) roof panel hypotheses. This results in the detection percentage almost doubling as about half the roof area in the scene is recovered. The 'Sparse Rural' image demonstrates that grouping is particularly important for more complex roof geometries and that the simple approach adopted is viable for A-frame roofs.

One final point to note is that even though the roof material of the buildings in the 'Sparse Rural' image appears to be uniform, the final hypotheses still arise from a number of different stack levels, as illustrated in Figure A.8, although some levels are absent and the original image does not contribute. The contribution of multiple stack levels to the final interpretation occurs because, although the roofs may appear to the human eye to be fairly homogeneous, the strict threshold used in the homogeneous operator only allows for a partial extraction of the roof panels at finer scales. As the scale increases, fully extracted panels may appear and persist over several scales if the panel's contrast with the background is marked. However, the fully extracted panel regions usually exhibit subtle differences along their boundaries, which translates into slightly different noise-free boundaries and regularised boundaries. These distinct boundaries have small differences in features which affects the scale at which detection occurs.

The shape accuracy metrics for the test images are given in Table 6.5. The lowest shape accuracies occur for the 'Imizamo Yethu' image, while the highest accuracies are produced for 'Sparse Rural'. These results mirror the area detection percentages presented earlier, for much the same reasons. The detection accuracy is greater than 59% in each image, however, for images other than 'Imizamo Yethu', it is substantially higher. In other words, whenever a shack or building is detected, on average, more than 59% of the roof area is detected.

The stricter QP shape accuracy measure (which reduces the accuracy based on the amount of non-building area found along with each building found) produces values within 10% of the DP measure for all images, and is greater than 50% for all images. The fact that the DP and QP averages are close implies that the false positive error per extraction is small.

The shape accuracy measure given in Equation 6.1 and used in [61] gives results that are close to the QP accuracy (within $\approx 5\%$). This measure produces accuracies which are sometimes greater than the QP accuracy and sometimes less than it. However, as explained earlier, this measure is less accurate than the QP measure.

6.7 System Parameters

As with all building detection systems, the different stages of this system rely on a number of parameters. The parameters that are specific to each image are the user defined area range, the shadow threshold and sun vector (Section 5.1.1). The remaining parameters are constant or image-invariant and are described in Appendix B. Additional image-invariant parameters, not included in Appendix B, are the number of scale-space stack levels and the number of iterations required to produce these levels (Table 5.1) as well as the shapes of the fuzzy membership functions and the fuzzy rule base (Sections 5.6.3 and 5.6.3).

It is difficult to envision theoretically optimal choices for the image-invariant parameters given the lack of a formal computer vision framework and the complexity of the building detection task. It is conceivable, but impractical, to determine parameters which optimise one or more of the performance measures for the entire test dataset through exhaustive search. Parameters optimised in this manner will, however, not be optimal for new images.

Therefore, the image-invariant parameters have been determined empirically, using

heuristics and keeping simplicity in mind. For example, if an extracted region overlaps with identified shadow it is excluded from further analysis. It is reasonable to allow a small amount of overlap as shadows have been dilated, with the result that a roof region casting a shadow may overlap with the shadow slightly. However, if too much overlap is allowed, shadow regions will be inadvertently detected. From empirical tests it appears that an allowable overlap area of close to 15% is a good choice and, for simplicity, a parameter value of exactly 15% is chosen. The number of scale-space stack levels used in the detection strategy, and the iterations required to produce them, as well as the fuzzy membership functions, have all been determined experimentally. A more rigourous approach to determining these parameters is suggested in Section 7.3.

A key issue to consider with image-invariant parameters is the sensitivity of system performance with respect to these parameters under changing scene and imaging conditions (sun vector direction and length and image contrast). The four images in Table 6.4 were acquired under three different sets of imaging conditions ('Marconi Beam 1' and '2' were acquired under the same set of conditions), contain both informal and formal housing ('Sparse Rural'), and include roofs with a variety of appearances. Well over a hundred shacks and buildings have been identified in the test imagery which indicates, within the limits of the test dataset, that the system has a degree of robustness.

The 'Imizamo Yethu' image does, however, demonstrate that low image contrast and the presence of vegetation negatively affects performance. Low image contrast could potentially be addressed by adjusting certain parameters, such as the homogeneity threshold, but it is felt that it would be better to either introduce a pre-processing stage responsible for increasing contrast, such as Wallis filtering [6], or attempt to acquire the source imagery under more suitable lighting conditions. The system is not designed to deal with rooftops which are occluded by vegetation, so this shortcoming cannot be addressed through the variation of system parameters.

All in all, the system works fairly well on different scenes given a mostly fixed set of parameters. This may seem surprising granted that there are distinct differences in the appearance of buildings and shacks within images and across images. The system's insensitivity to parameter choices is a direct consequence of adopting a scale-space approach. Each source image is deliberately transformed in a manner which reduces the differences in the appearance of the structures being detected. Also, although the parameters for region extraction are fixed, the images to which they are applied vary as the scale changes. In other words, parameter variation has

been replaced to some degree by image variation over scale. Using a scale-space reduces the likelihood of making parameter choices which work extremely well for one particular image and less so for others because, at the very least, the parameters chosen need to enable the detection of the target objects from a family of images rather than from a single source image.

6.8 Comparison to Existing Systems

Table 6.6 presents the performance of shack detection systems as reported in the literature by various researchers. Each of the approaches listed is described more fully in the literature survey (Section 3.3). The table is restricted to papers and theses giving quantitative results. Only detection percentages and shape accuracy are compared. When determining building counts, it is assumed that a building or shack is detected if any part of its roof is detected. Quality percentages are not included as these have not been provided in many cases.

Approach	Source Data	Images Tested
Li (2000) [13]	Aerial orthoimage and derived DSM	1*
Mayunga et al. (2007) [12]	Quickbird (satellite) imagery	2
Baltsavias et al. (1997) [79]	Aerial orthoimage and derived DSM	1*
Rüther et al. (2002) [61]	Aerial orthoimages and derived DSMs	3*
This work	Aerial orthoimages and unrectified images	4*

Approach	Automation	Building Count DP	Area DP	Shape Acc (Equation 6.1)
Li	Semi-auto (shadow edges)	94	58	57
	Semi-auto (shadow, DSM blob edges)	100	81	73
Mayunga	Semi-auto (snakes)	100†	-	-
Baltsavias	Auto (DSM blob boundaries)	100	67	-
Rüther	Auto (snakes from DSM blob centres)	62 (62)	-	81 (79)
This work	Auto (scale-space, shadow verification)	92 (94)	70 (70)	67 (61)

Table 6.6: Reported performances (%) for shack detection systems, where available. *The 'Marconi Beam 2' image formed part of the test dataset. For systems which have been tested on multiple images and the test dataset includes the 'Marconi Beam 2' image, the average performance measure for all images is given, followed in parentheses by the measure for the 'Marconi Beam 2' image.
†Not stated directly but all buildings are completely or partially extracted.

It is difficult to compare shack detection systems in an unambiguous fashion due to the lack of standard datasets (and the corresponding ground truths), differing system goals and different measures of performance (Section 2.8). A particular issue for these systems is the variation in architecture of settlement buildings in different countries (see Section 3.3) which may invalidate many of the assumptions made in one or other system. This work is focused on shack detection in scenes of typical South African informal settlements. A number of researchers [79, 13, 61] working in this area have used the 'Marconi Beam 2' image as part of their datasets. This allows for a fairly direct comparison to be made to these systems, albeit only for a single image and in spite of differing system goals (automated versus semi-automated) and source data.

In [79, 13, 61] colour imagery is required as opposed to the greyscale imagery used here because shadow is detected through PCA analysis of the colour bands in the source image (see [13] for more detail). Stereo image matching is used in deriving the DSMs and it is assumed that a DTM of the area being modelled exists in order to generate orthoimages. The system presented here does not require height data as an input. The Quickbird imagery utilised in [12] is panchromatic with a resolution of 0.6 m, which is considerably lower than the resolution of the images used in all of the other systems, including this one.

The approaches documented in [13] and [12] are semi-automated in that user interaction is required for the initial identification of shacks. For semi-automated systems, it is to be expected that the building count DP should be 100% as a user is manually identifying the buildings. This is borne out by the percentages reported in Table 6.6. Note, the building count DP may not always be 100% if there is insufficient supporting evidence to originate a hypothesis from the click point. The area DP increases if DSM (height) information is used as shown in [13]. The shape accuracy also improves with height data. However, height data is not used in [12] and a high mean shape accuracy of 91% is reported (using the slightly different measure of area coverage rate).

Results from automated systems have been reported in [79, 61]. In [79] DSM blobs are used to *coarsely* delineate shacks resulting in a 100% building count DP. When shacks are closely clustered, however, the blobs tend to span multiple shacks. Additionally, many of the shack boundaries are poorly localised. In [61] snakes are automatically initialised on the image and their position is optimised to delineate shack boundaries. The building detection rate is relatively low using this method due to large numbers of false negatives which appear to originate from inaccuracies in the DSM. The shape

accuracy, however, is relatively high, despite the fact the boundaries appear to be poorly localised.

The system presented here is capable of achieving building detection rates of over 92% on the test imagery. This is close to the performance of semi-automated systems and the automated system in [79], and is significantly better than the system in [61]. The area DP is relatively high with an average detection rate of 70%. For the 'Marconi Beam 2' image specifically, the detection rate is also 70% and this is only exceeded by Li's semi-automatic approach, which uses shadows and DSM blob edges.

In terms of shape accuracy the performance is mediocre when compared with the other systems. Rüther et al's system has a shape accuracy of 79% for the 'Marconi Beam 2' image which is significantly better than the 61% achieved here. However, these shape accuracy figures are difficult to compare directly because of the large difference in building count detection percentages (over 30%). This system detects a far greater percentage of buildings in the scene. This may imply that the additional shacks detected (over and above the ones detected by Rüther et al's system) are inherently more difficult to delineate correctly. Shape accuracy can only be directly compared if the comparison is limited to the shacks identified by both systems.

A number of building detection systems, in which closed boundary localisation and regularisation techniques play an important part, have been reviewed in Chapter 3 and are summarised in Tables 3.1 and 3.2. Only the systems described in [61] and [12] attempt to localise closed boundaries for shack delineation in informal settlement imagery. The localisation performance of this system is visibly better than [61] and similar to [12], with the localisation of the final boundaries ranging from good to mediocre. There are qualitative differences, however, in the results. In [12] corner points tend to be rounded. This is presumably due to the absence of snake energy terms for enforcing geometric constraints, such as near-parallelism and orthogonality. In this system, corner points are not rounded and hypotheses appear more rectilinear.

Here, there may be gross errors if the starting boundary is poorly localised (in spite of attempts to filter out such boundaries). In a semi-automated system user intervention is used to eliminate poorly localised starting boundaries and these will not appear in the final results. For example, in [12], a user may reject the originating snake seed point and its associated snake contour, prior to snake optimisation, if the contour is deemed to be ill-fitting.

Finally, it is worth noting that in previous work done by the author the 'Sparse Rural' image has been tested using a non-scale-space system based on the homogeneous

operator [115]. However, the method of shadow verification is quite different, so even though the detection results of this system are substantially better, the results cannot be directly compared.

6.9 Conclusion

The performance of the shack detection system which embodies the detection strategy presented in Chapter 5 is thoroughly analysed for a number of test images. A qualitative assessment is conducted through visual inspection of the system's results overlaid on the relevant ground truth. This demonstrates the accuracy of the system visually and allows one to see the rectilinear nature of the hypotheses that are produced. Standardised quantitative metrics are provided, such as the detection and quality percentages for both building counts and area, and the shape accuracy. Together, these metrics offer greater insight into the overall performance of the system and demonstrate that it is capable of identifying the majority of shacks and buildings in the test imagery.

The system is additionally evaluated on a per-stage and per-boundary-type basis. From this analysis the utility of the different stages is shown as well as the effectiveness of the model-driven-simplification and expansion techniques. The scale-space stack is found to be a useful construction as the hypotheses that best correspond to shacks or buildings are drawn from many different levels of the stack. The execution time of each of the stages is compared and suggestions, such as parallelising the analysis of individual images in the stack, are offered for reducing the overall time taken. The results of this system are compared to other systems, particularly for the 'Marconi Beam 2' image. This system is found to perform well when compared to automated systems and adequately when compared to semi-automated systems. Additionally, if DSM data is not used then this system exceeds the performance of the semi-automated system described in Li [13] when operating on the 'Marconi Beam 2' image.

The following chapter presents the conclusions of the study and recommendations for future work are given.

Chapter 7

Conclusion

7.1 Conclusions

The detection strategy presented is based on the construction of an anisotropic scale-space from a single source image. Homogeneous regions and high contrast edges in the source image are preserved as the image is blurred over scale. Intermediate-contrast edges are removed over scale and when these correspond to roof substructure boundaries, the shack roof of which the substructure is a part, is emphasised. Homogeneous regions are identified across all scales and the boundaries of these regions are simplified to remove digitisation noise. These boundaries form the initial set of hypotheses. These hypotheses are verified across all scales using shadow and are regularised in accordance with an implicit building model. A fuzzy rule base is used to gauge the strength of the supporting evidence for each hypothesis. Conflicting (overlapping) sets of hypotheses are resolved by selecting the hypothesis with the strongest supporting evidence in each set. Grouping, boundary expansion and a final verification of hypotheses results in the system's output.

This multi-scale strategy has been designed specifically for identifying shacks in informal settlement scenes (see 4.5). Within this scope the system is shown to work, and work well, in that it is capable of achieving better performance in certain respects than both semi-automated systems which do not utilise height data and automated systems which do use height data. The fact that is is possible to achieve the same ballpark performance as an existing automated system using less rich source data (a single intensity image) is especially significant, and highlights the value of the strategy. This system is less capable of dealing with scenes in which shacks are obscured by vegetation and in which image contrast is poor.

It has also been shown that this strategy can work well in other contexts, such as rural formal settlements with buildings of simple architecture. However, the system's performance is expected to degrade rapidly when applied to images further removed from the design scope, such as lower resolution images, scenes containing buildings with more complex architecture, scenes of extremely dense informal settlements, urban environments, and so on. The system's performance in these areas has not been investigated as it is deemed to be beyond the scope of this study.

The system as presented is usable, with a little corrective editing, for a certain class of end-user applications such as shack counting, electrical reticulation planning and settlement monitoring. These types of applications often do not require accurate shack boundaries (and the model-driven simplified boundaries may even suffice). For example, in electrical reticulation planning only the approximate centroid of each shack is required. For applications which do require accurate shack boundaries, such as a service for routing an emergency vehicle to a particular area of a settlement, the corrective-editing step will be more substantial. Nonetheless, as other studies have shown, automating as much of the task as possible results in a significant time-saving compared to completely manual delineation.

This study has argued that when performing shack detection on informal settlement scenes, simplification of the source imagery is a viable approach to dealing with the problems presented by substructure. The intent behind this approach is to discard fine-scale detail which is not relevant, and complicates the detection and delineation process. It has been shown that a multi-scale representation of shacks homogenises their appearance. This allows for a detection strategy based on a single, simple object model which is capable of achieving good results when applied to a family of simplified images derived from a single source image. Not only are source images blurred through anisotropic diffusion but hypotheses' boundaries are also simplified through discrete curve evolution in order to remove digitisation noise. This abstraction process has utility in that it allows shape measures to be reliably calculated.

Both of the simplification techniques which have been selected do not dislocate the features of the underlying image or curve over scale, unlike Gaussian scale-spaces. This ensures that the homogeneous regions derived from different scales remain well localised and can be successfully integrated with edges derived from the source image in the boundary expansion stage. Additionally, the rectilinearity and compactness measures are not compromised during the digitisation noise removal process. This would not be the case, if, for example, a Gaussian-based curve simplification technique

was used, as corners would be rounded.

Sohn & Dowman [29] observe, with respect to building detection in aerial and satellite images, that the cues for building reconstruction provided by low-level feature extractors are insufficient. Furthermore, they note that in an attempt to overcome this:

- Single cues in 2D have been enhanced "physically" in quantity and quality by using multiple images of the scene from different angles.
- Additionally, "cue integration was performed, rather than enhancing a single cue's property 'physically', by combining it with other multiple cues 'conceptually', such as color constancy, brightness homogeneity and texture regularity" [29, p. 346].

This work illustrates a third approach to cue enhancement — cues can be enhanced by producing multiple versions of the same image of the scene at different scales. This approach to cue enhancement, as with cue integration, does not rely on richer source data.

It is interesting to contrast this multi-scale strategy with that presented by Jin & Davis [35]. In Jin & Davis's work, a morphological scale-space is used based on a pixel's differential morphological profile. A key feature of this scale-space is that it enables the extraction of structures of differing sizes based on the size of the structuring element. Urban scenes contain buildings of a variety of sizes and, hence, this is an important concern. In this work, as the scale increases, the image is increasingly smoothed promoting intra-region homogeneity, which increases the size of the regions extracted. However, the goal is not to extract shacks at multiple sizes — as most shacks are quite similar in size — but to extract complete shacks rather than roof substructure. In [35] substructure is ignored by using structuring elements designed for extracting medium to large buildings only (small buildings, often similar in size to substructure, are identified using an alternative approach). In this work substructure is blurred out of existence over scale and verification procedures are used to prevent substructure which has been extracted at finer scales from appearing in the final interpretation of the scene.

In the strategy presented, there is an interplay between model- and data-driven control. Initially, the strategy is data-driven as homogeneous regions are extracted at all scales. However, at various points in the strategy, the object model drives the simplification of the region boundaries which, in turn, dictates where the image is

sampled for supporting evidence or constrains the search for edges. This is typical of hypothesise-and-verify approaches. The focus of this work is largely on the development of mid-level processes within the context of a region-based scale-space approach.

Shadow plays a vital role in this system, as in many others, by providing a 3D height cue which is used in verifying hypotheses[1]. It is therefore, essential that the source imagery is acquired when shadow is present. As stated in [61], it is important to let the image acquisition process be informed by the extraction strategy adopted. The exaggeration of shadow regions, through dilation, aids the detection process in circumstances where shacks are closely clustered and shadows are occluded. This approach is viable if shadow is not used for precise calculations, such as determining shack heights or boundaries.

An automated approach based on initially identifying homogeneous *regions* is a sound strategy for the interpretation of informal settlement imagery within the context of a scale-space, as it can be expected that shack roofs (or parts of the roofs) will become more uniform in appearance over scale. Importantly, adopting a region-based approach results in closed boundaries and avoids the problems of edge fragmentation which edge-based approaches are subject to. Additionally, these regions are derived through thresholding and further processing focuses on these regions and their immediate surroundings only. This is different to many region-based segmentation techniques in which the entire image is segmented and still has to be processed in subsequent stages.

However, a region-based approach is not a panacea for shack detection as boundary regularisation and localisation now assume a very important role. Special care has to be taken in dealing with digitisation noise which affects region boundaries. This noise invalidates shape measures, such as rectilinearity, and has to be removed before reliable shape measures can be calculated.

The problem of edge fragmentation is replaced by one of region flooding. This can be addressed by imposing a desired building shape on the region boundary through eliminating selected vertices. However, this imposed shape will only correspond to genuine shack borders if a sufficient portion of the initial boundary is well localised so that the canonical orientation of the shack can be correctly determined. In

[1]Note, that in existing shack detection systems, shadow is not used for this purpose as DSM data is available, or a user is manually performing the identification. Still, even in most of these systems, shadow is important for narrowing the search space by masking regions of "non-interest" or providing delineation cues.

other words, it is important to make use of a model-driven simplification process that incorporates global shape features (canonical orientation, rectilinearity and compactness) in addition to local features (vertex relevance), as local features may be highly inaccurate at points.

Region reference boundaries sit slightly within the borders of the rooftops being detected. Localisation of these boundaries can be improved by using appropriate edge information. The search for relevant edges can be constrained, given knowledge of their expected location and orientation based on the reference boundary. Other shack detection strategies use DSM blobs or user clickpoints to identify regions of interest in an image, and limit the search for image evidence. Purely edge-based approaches lack this knowledge and edge selection and grouping becomes a major issue in its own right. This would be particularly true for informal settlement imagery in which many spurious edges are present on shack rooftops.

A simple 2D building (object) model — a four- to six-sided compact, rectilinear polygon — can be usefully employed to detect the majority of shacks and buildings within informal settlements[2] which reinforces existing research. Note, that this model is implicitly encoded in the model-driven simplification algorithm (while the fuzzy rules represent in a more declarative fashion the criteria for hypothesis selection). Shape is a crucial element of this model and Žunić & Rosin's fairly recently published rectilinearity measure has been used extensively in this system. It is important that model constraints are "relaxed" in that geometric properties such as rectangularity, parallelism and orthogonality are not strictly required of, or enforced upon, the shack outline. This is essential for modelling informal settlement buildings as it cannot be expected that these buildings will have been constructed with the same geometrical regularity as those in formal settlements.

Separating the phases of boundary regularisation, and boundary localisation and re-regularisation, allows edge evidence to be processed effectively. During the model-driven simplification process hypotheses' boundaries are regularised. At this stage no image data is used and the regularisation takes place through applying an implicit shack model which optimises rectilinearity and compactness. This allows boundaries to be correctly recovered and unaligned spurious edges to be ignored during the boundary expansion stage. Snakes are often used for integrating the processes of regularisation and localisation. This is done by including energy terms for both enforcing (strong or weak) geometrical constraints and for attracting the snake

[2]The model is also suitable for delineating the footprint of buildings with non-flat roofs which are composed of rectangular planes, such as A-frame roofs.

contour to image edges. That is, a snake's movement is governed by an attempt to both localise and regularise the contour. One of the challenges in applying snakes to informal settlement imagery is the fact that strong spurious edges located within the neighbourhood of the snake can interfere with the localisation aspect and produce poor results. It is possible to trade-off localisation and regularisation forces by using appropriate weightings but it is difficult to selectively use edge evidence within a snake model formulation.

The localisation process is capable of dealing with fragmented edge evidence by inferring corner points from pairs of fragments. The building model is then re-applied to the expanded boundary and is fairly successful in eliminating false corners. In [94] the accuracy of the final boundary is shown to be sensitive to the orientation of the initial segmented region on which it is based. Here, this is ameliorated by including new image evidence in the form of edges and allowing a tolerance with which these edges are required to align with the reference boundary. This approach, within limits, allows for accurate boundary localisation in spite of an initially mis-oriented segmentation.

Resolving competing hypotheses across different scales is not a trivial problem. The approach taken is to regard hypotheses as mutually exclusive and select the hypothesis with the strongest evidential support from each competing set. For small and medium hypotheses, the evidence is not that conclusive with respect to their likelihood but for larger hypotheses the decision is more clear cut. The ability to non-linearly combine and reason about the supporting evidence in these circumstances is useful, and achieved through the use of a fuzzy rule system.

Sections of a shack roof may be so different in colour/texture from each other that they are extracted as distinct regions, and consequently, hypotheses, over all scales. In these cases, the entire roof may still be recovered by grouping non-shadow-verified hypotheses adjacent to shadow-verified hypotheses (along the direction of the sun vector). This has been successfully demonstrated and does improve the final results, although the improvement is small. For non-flat roofs, composed of rectangular planes (such as A-frame roofs), this grouping procedure becomes more of a necessity, especially, if a strong contrast exists between the roof panels. A more sophisticated grouping approach would be required for roof surfaces of higher complexity.

Performing a large part of the interpretation using a vector-based representation of points and lines rather than pixel-based (raster) representation affords one the opportunity to make use of techniques with greater precision. For example, the image

can be sampled directly along the sun vector and accurate shape measures can be used.

7.2 Summary of Contributions

A novel strategy has been developed for automating the detection and delineation of informal settlement buildings within the context of an anisotropic scale-space. To the knowledge of the author, this is the first known application of an *anisotropic scale-space* to the building detection problem in general – not just shack detection[3]. This strategy is a unique and valuable addition to the toolbox of techniques that can be employed to automate the interpretation of informal settlement imagery and minimise user intervention. Additionally, this is the first automated system capable of extracting individual shacks which is not reliant on DSM height data in any respect during the detection process.

A number of specific research contributions have been made, including:

- It has been demonstrated that having a distinct boundary regularisation phase followed by a localisation/re-regularisation phase proves to be valuable. The contribution of these different phases is illustrated by the DP and QP metrics achieved per boundary type. The success of this approach reinforces previous work[4] and shows that it is also valuable for informal settlement scenes. It is possible that a separate regularisation phase (and therefore a strong reliance on the object model) might be even more necessary for informal settlement scenes where patterns in the image evidence are less predictable.
- Novel algorithms for the model-driven simplification of shack roof boundaries have been developed which are based on simultaneously trying to optimise the rectilinearity and compactness measures of such boundaries through vertex removal. In these boundary regularisation algorithms, global geometric constraints, like parallelism, are not enforced in order to take into account the special characteristics of informal settlements. Experimental results demonstrate that these algorithms are capable of correcting fairly large boundary distortions. The model-driven simplification technique is simple to implement

[3] Although, there are building detection systems which are based on other types of scale-spaces and scale-space-like hierarchies.

[4] In [87] the regularisation and localisation techniques (which are different to those used here) are decoupled providing good results for scenes of industrial buildings.

and works well for the compact, rectilinear shapes prevalent in this domain. It is not as general as some of the other regularisation techniques reviewed.

- The manner in which homogeneous regions are linked across scales is similar in spirit to the way features are linked in other scale-space systems. However, viewing these regions as competing hypotheses and using a fuzzy system for resolving conflicts over scale (rather than from a single scale) appears to be unique.
- A new technique for determining the stopping point in discrete curve evolution is presented. This technique is intuitive, simple to implement, and has been shown to be robust over scale.
- A homogeneous operator has been designed for identifying homogeneous regions at each scale. This segmentation strategy incorporates domain knowledge as shack roofs are expected to assume a uniform appearance over scale.

7.3 Recommendations for Future Research

The detection strategy presented does not make use of height data as a single intensity image is all that is required to produce the anisotropic scale-space. Consequently, shadow forms an important cue for verifying 3D structures. Incorporating height data, when it is available, should increase detection rates as this data can assist in 3D structure verification. The approach presented here can be usefully combined with systems which do make use of DSMs, and investigating exactly how this integration would occur would be worthwhile.

In this system fuzzy membership functions have been handcrafted, although some aspects are based on an understanding of the domain. This has been a fairly laborious process and there is no guarantee that the final sets produce optimal results in some sense (there is a tradeoff between detection and quality percentages). It has been shown in [71] that when comparing a handcrafted system with automated learning methods, the automated methods perform better over most of the operating range. Therefore, it is felt that by adopting a more systematic approach to evaluating evidence, which incorporates machine learning techniques for shaping the fuzzy sets (and possibly deriving the fuzzy rules), a better tradeoff position may be achievable.

More investigation is required into the deep structure of informal settlement images. It would be interesting to monitor how the features of hypotheses evolve over scale. It may be that genuine hypotheses demonstrate some behaviour in their evolution

which distinguishes them from false hypotheses and that this can be used as an additional verification aid. Alternatively, analysing behaviour over scale might help in determining where background flooding occurs, as there will be a marked increase in hypothesis area potentially accompanied by a change in canonical orientation. This knowledge could be used in shape recovery. Finally, scale-space behaviour could be used to attach a confidence value to each hypothesis in the final interpretation. This value could be based on the stability of the hypothesis over scale, the number of unverified hypotheses lying on the same path in the scale-tree and so on.

The current approach to generating the image stack relies on a fixed number of stack levels and iterations per level which have been experimentally determined (Section 5.2.3). It may be worth adapting these factors on a per-image basis. The diffusion stopping point could be determined by considering the rate of change in the number of regions extracted. When the rate decreases to an acceptable level, indicating that further iterations will produce very few mergers, then diffusion can be stopped. Alternatively, the stopping point could be determined by setting a lower limit on the ratio of regions extracted at the current level to those extracted from the original image.

In order to sample the scale-space adequately, it is important to set a limit on the increase in extracted area that occurs from one stack level to the next. If the increase is too large, important scale-space merge events may be missed. It is envisioned that during the production of the scale-space, the experimentally determined values given in Table 5.1 would form estimates of the number of iterations to apply per level. Once an image is derived at a given level, using the estimated number of iterations, the homogeneous area would need to be extracted and compared to the area extracted from the previous level. If the growth in area has exceeded the limit then an intermediate (less diffuse) level would need to be created based on a smaller number of iterations. A scheme like this would ensure that the intervals at which the scale-space is sampled are adapted to the source image, at the cost of being more computationally expensive.

References

[1] Mason, S.O. & Fraser, C.S. (1998), 'Image sources for informal settlement management', *The Photogrammetric Record*, vol. 16, no. 92, pp. 331–345.

[2] UN-HABITAT, U.N.H.S.P. (2003), *The Challenge of Slums. Global Report on Human Settlements 2003*, Earthscan Publications Ltd, London and Sterling.

[3] Abbott, J. & Douglas, D. (2003), 'The use of longitudinal spatial analyses of informal settlements in urban development planning', *Development South Africa*, vol. 20, no. 1.

[4] Sietchiping, R. (2004), *A Geographic Information Systems and Cellular Automata-Based Model of Informal Settlement Growth*, Ph.D. thesis, School of Anthropology, Geography and Environmental Studies, The University of Melbourne, Australia.

[5] Li, J., Li, Y., Chapman, M.A. & Rüther, H. (2005), 'Small format digital imaging for informal settlement mapping', *Photogrammetric Engineering and Remote Sensing*, vol. 71, no. 4, pp. 435–442.

[6] Mason, S. & Baltsavias, E. (1997), 'Image-based reconstruction of informal settlements', in [20], pp. 97–108.

[7] Abbott, J. (2003), 'The use of GIS in informal settlement upgrading: Its role and impact on the community and on local government', *Habitat International*, vol. 27, pp. 575–593.

[8] Hofmann, P., Strobl, J., Blaschke, T. & Kux, H. (2006), 'Detecting informal settlements from QuickBird data in Rio De Janeiro using an object based approach', in 'Proceedings of the ISPRS, Commission IV', Goa, India.

[9] Hofmann, P. (2001), 'Detecting informal settlements from IKONOS image data using methods of object oriented image analysis: An example from Cape Town (South Africa)', *Remote Sensing of Urban Areas / Fernerkundung in urbanen Räumen*, vol. 35, pp. 107–118.

[10] Seang, T.P., Mund, J.P. & Symann, R. (Date last accessed: 08/04/2009), *Low Cost Amateur Aerial Pictures with Balloon and Digital Camera - Practitioner's Guide*, MethodFinder, `http://www.methodfinder.net/briefdescription81.html`.

[11] Barry, M. & Rüther, H. (2001), 'Data collection and management for informal settlements upgrades', in 'Proceedings of the International Conference on Spatial Information for Sustainable Development', Nairobi, Kenya.

[12] Mayunga, S.D., Coleman, D.J. & Zhang, Y. (2007), 'A semi-automated approach for extracting buildings from QuickBird imagery applied to informal settlement mapping', *International Journal of Remote Sensing*, vol. 28, no. 10, pp. 2343–2357.

[13] Li, J. (2000), *Informal Settlement Modeling Using Digital Small-Format Aerial Imagery*, Ph.D. thesis, Department of Geomatics, University of Cape Town, Cape Town, South Africa.

[14] Ünsalan, C. & Boyer, K.L. (2005), 'A system to detect houses and residential street networks in multispectral satellite images', *Computer Vision and Image Understanding*, vol. 98, pp. 423–461.

[15] de la Rey, A. (2006), 'Personal communication', Geographical Information Manager ESKOM ESI-GIS.

[16] Argialas, D.P. & Harlow, C.A. (1990), 'Computational image interpretation models: An overview and perspective', *Photogrammetric Engineering and Remote Sensing*, vol. 56, no. 6, pp. 871–886.

[17] Mayer, H. (1999), 'Automatic object extraction from aerial imagery–a survey focusing on buildings', *Computer Vision and Image Understanding*, vol. 74, no. 2, pp. 138–149.

[18] Crevier, D. & Lepage, R. (1997), 'Knowledge-based image understanding systems: A survey', *Computer Vision and Image Understanding*, vol. 67, no. 2, pp. 161–185.

[19] Gruen, A., Kuebler, O. & Agouris, P. (editors) (1995), *Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Birkhäuser Verlag, Basel.

[20] Gruen, A., Baltsavias, E.P. & Henricsson, O. (editors) (1997), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, Birkhäuser Verlag, Basel.

[21] Baltsavias, E.P., Gruen, A. & Gool, L.V. (editors) (2001), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*, Balkema Publishers, Rotterdam.

[22] Gruen, A. & Nevatia, R. (editors) (1998), *Special Issue on 'Automatic Building Extraction from Aerial Images', Computer Vision and Image Understanding*, vol. 72, no. 2.

[23] Remondino, F. (Date last accessed: 08/06/2006), 'ISPRS - international society for photogrammetry and remote sensing', `http://www.isprs.org/isprs.html`.

[24] Baltsavias, E.P. (2004), 'Object extraction and revision by image analysis using existing geodata and knowledge: Current status and steps towards operational systems', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 58, pp. 129–151.

[25] Sowmya, A. & Trinder, J. (2000), 'Modelling and representation issues in automated feature extraction from aerial and satellite images', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 55, pp. 34–47.

[26] Guindon, B. (1997), 'Computer-based aerial image understanding: A review and assessment of its application to planimetric information extraction from very high resolution satellite images', *Canadian Journal of Remote Sensing*, vol. 23, no. 1, pp. 38–47.

[27] Marr, D. (1982), *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*, W H Freeman and Company, San Francisco.

[28] Förstner, W. (1995), 'Mid-level vision processes for automatic building detection', in [19], pp. 179–188.

[29] Sohn, G. & Dowman, I.J. (2001), 'Extraction of buildings from high resolution satellite data', in [21], pp. 345–356.

[30] Ballard, D.H. & Brown, C.M. (1982), *Computer Vision*, Prentice-Hall, New York, first edn.

[31] Hanson, A.R., Marengoni, M., Schultz, H., Stolle, F., Riseman, E.M. & Jaynes, C. (2001), 'Ascender II: a framework for reconstruction of scenes from aerial images', in [21], pp. 25–34.

[32] Huertas, A. & Nevatia, R. (1988), 'Detecting buildings in aerial images', *Computer Vision, Graphics and Image Processing*, vol. 41, pp. 131–152.

[33] Heipke, C., Jacobsen, K. & Mills, J. (2006), 'Editorial for the theme issue: "digital aerial cameras"', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 60, pp. 361–362.

[34] Muller, J.P., Ourzik, C., Kim, T. & Dowman, I. (1997), 'Assessment of the effects of resolution on automated DEM and building extraction', in [20], pp. 233–242.

[35] Jin, X. & Davis, C.H. (2005), 'Automated building extraction from high-resolution satellite imagery in urban areas using structural, contextual and spectral information', *EURASIP Journal on Applied Signal Processing*, vol. 14, pp. 2196–2206.

[36] Kim, T. & Muller, J.P. (1999), 'Development of a graph-based approach for building detection', *Image and Vision Computing*, vol. 17, pp. 3–14.

[37] Paparoditis, N., Cord, M., Jordan, M. & Cocquerez, J.P. (1998), 'Building detection and reconstruction from mid- and high-resolution aerial imagery', *Computer Vision and Image Understanding*, vol. 72, no. 2, pp. 122–142.

[38] Baltsavias, E. & Hahn, M. (1998), 'Cooperative algorithms and techniques of image analysis and GIS', in 'Proceedings of ISPRS Commission IV, Working Group IV/III.2', Stuttgart, Germany.

[39] Vosselman, G. (2002), 'Fusion of laser scanning data, maps, and aerial photographs for building reconstruction', in 'Proceedings of IGARSS'02: IEEE International Geoscience and Remote Sensing Symposium and the 24th Canadian Symposium on Remote Sensing', Toronto, Canada.

[40] Fraser, C.S., Baltsavias, E. & Gruen, A. (2002), 'Processing of IKONOS imagery for submetre 3D positioning and building extraction', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 56, pp. 177–194.

[41] McKeown Jr, D.M. (1996), 'Top ten lessons learned in automated cartography', in 'Proceedings of the 1996 ARPA Image Understanding Workshop', Palm Springs, California.

[42] Jaynes, C., Riseman, E. & Hanson, A. (2003), 'Recognition and reconstruction of buildings from multiple aerial images', *Computer Vision and Image Understanding*, vol. 90, pp. 68–98.

[43] Heuel, S. & Kolbe, T.H. (2001), 'Building reconstruction: The dilemma of generic versus specific models', *Künstliche Intelligenz*, , no. 3, pp. 57–62.

[44] Weidner, U. (1996), 'An approach to building extraction from digital surface models', in 'Proceedings of the 18th ISPRS Congress', Wien, vol. B3, pp. 924–929.

[45] Irvin, R.B. & McKeown, D.M. (1989), 'Methods for exploiting the relationship between buildings and their shadows in aerial imagery', *IEEE Trans. Systems, Man and Cybernetics*, vol. 19, no. 6, pp. 1564–1575.

[46] Shufelt, J.A. & McKeown, D.M. (1993), 'Fusion of monocular cues to detect man-made structures in aerial imagery', *Computer Vision, Graphics and Image Processing: Image Understanding*, vol. 57, no. 3, pp. 307–330.

[47] Bloch, I. & Maître, H. (1997), 'Data fusion in 2D and 3D image processing: An overview', in 'Proceedings of the Brazilian Symposium on Computer Graphics and Image Processing', Campos do Jordao, Brazil, pp. 127–134.

[48] Lin, C. & Nevatia, R. (1998), 'Building detection and description from a single intensity image', *Computer Vision and Image Understanding*, vol. 72, no. 2, pp. 101–121.

[49] Liow, Y. & Pavlidis, T. (1990), 'Use of shadows for extracting buildings in aerial images', *Computer Vision, Graphics and Image Processing*, vol. 49, no. 2, pp. 242–277.

[50] Gülch, E. (1997), 'Application of semi-automatic building acquisition', in [20], pp. 129–138.

[51] EPFL - LIG, Lausanne, Switzerland (Date last accessed: 08/04/2009), 'CROSSES website - CROowd Simulation System for Emergency Situation', `http://ligwww.epfl.ch/~thalmann/crosses.html`.

[52] Kim, Z. & Nevatia, R. (2004), 'Automatic description of complex buildings from multiple images', *Computer Vision and Image Understanding*, vol. 96, pp. 60–95.

[53] Scholze, S. (Date last accessed 28/06/2006), 'ETH Zurich - BIWI: AMOBE II', `http://www.vision.ee.ethz.ch/projects/Amobe_II/`.

[54] Shi, Z., Shibasaki, R. & Murai, S. (1997), 'Automated building extraction from digital stereo imagery', in [20], pp. 119–128.

[55] Apostolellis, J., Tumazos, S., West, N., Dwolatzky, B. & Meyer, A.S. (1996), 'The evolution of CAD-based optimisation tools for distribution network design', in 'Proceedings of the 4th IEEE Africon Conference', Stellenbosch, South Africa, vol. 1, pp. 496–499.

[56] Tumazos, S.J.C., Dwolatzky, B. & Meyer, A.S. (1996), 'A technique for automating stands to junction allocation in electrification design using heuristics', in 'Proceedings of the 6th South African Universities Power Engineering Conference (SAUPEC)', Johannesburg, South Africa, pp. 223–225.

[57] Chen, Z., Delis, A. & Bertoni, H.L. (2004), 'Building footprint simplification techniques and their effects on radio propagation predictions', *The Computer Journal*, vol. 47, no. 1, pp. 103–133.

[58] Baltsavias, E., Pateraki, M. & Zhang, L. (2001), 'Radiometric and geometric evaluation of IKONOS geo images and their use for 3D building modelling', in 'Proceedings of the Joint ISPRS Workshop on High Resolution Mapping from Space', Hanover, Germany.

[59] Letournel, V., Sankur, B., Pradeilles, F. & Maître, H. (2002), 'Feature extraction for quality assessment of aerial image segmentation', in 'Proceedings of the ISPRS Commission III Symposium: "Photogrammetric Computer Vision", part A', pp. 199–204.

[60] Shufelt, J.A. (1999), 'Performance evaluation and analysis of monocular building extraction from aerial imagery', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 311–326.

[61] Rüther, H., Martine, H.M. & Mtalo, E.G. (2002), 'Application of snakes and dynamic programming optimisation technique in modeling of buildings in informal settlement areas', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 56, pp. 269–282.

[62] Song, W. & Haithcoat, T.L. (2005), 'Development of comprehensive accuracy assessment indexes for building footprint extraction', *IEEE Trans. Geoscience and Remote Sensing*, vol. 43, no. 2, pp. 402–404.

[63] Henricsson, O. & Baltsavias, E. (1997), '3-D building reconstruction with ARUBA: A qualitative and quantitative evaluation', in [20], pp. 65–76.

[64] ETH Zürich (Date last accessed: 08/04/2009), 'Ascona workshop 2001', `http://www.photogrammetry.ethz.ch/news/events/ascona/ascona2001.html`.

[65] Canny, J. (1986), 'A computational approach to edge detection', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698.

[66] Burns, J.B., Hanson, A.R. & Riseman, E.M. (1986), 'Extracting straight lines', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 8, no. 4, pp. 425–455.

[67] Boldt, M., Weiss, R. & Riseman, E. (1989), 'Token-based extraction of straight lines', *IEEE Trans. Systems, Man and Cybernetics*, vol. 19, no. 6, pp. 1581–1594.

[68] Venkateswar, V. & Chellappa, R. (1992), 'Extraction of straight lines in aerial images', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, pp. 1111–1114.

[69] Mohan, R. & Nevatia, R. (1989), 'Using perceptual organisation to extract 3-D structures', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 11, pp. 1121–1139.

[70] Lin, C., Huertas, A. & Nevatia, R. (1994), 'Detection of buildings using perceptual grouping and shadows', in 'Proceedings of the IEEE Computer Vision and Pattern Recognition Conference', pp. 62–69.

[71] Kim, Z. & Nevatia, R. (1999), 'Uncertain reasoning and learning for feature grouping', *Computer Vision and Image Understanding*, vol. 76, no. 3, pp. 278–288.

[72] Jaynes, C., Stolle, F. & Collins, R. (1994), 'Task driven perceptual organisation for the extraction of rooftop polygons', in 'Proceedings of the ARPA Image Understanding Workshop', pp. 359–365.

[73] Krishnamachari, S. & Chellappa, R. (1993), 'Delineating buildings by grouping lines', Technical Report CS-TR-3127, Computer Vision Laboratory: Center for Automation Research, University of Maryland.

[74] Cooper, B.E., Chenoweth, D.L. & Selvage, J.E. (1994), 'Fractal error for detecting man-made features in aerial images', *Electronics Letters*, vol. 30, no. 7, pp. 554–555.

[75] Priebe, C.E., Solka, J.L. & Rogers, G.W. (1994), 'Discriminant analysis in aerial images using fractal based features', Technical report, Naval Surface Warfare Centre, Dhalgren, Virginia 22448.

[76] Levitt, S.P. & Aghdasi, F. (1997), 'Texture measures for building recognition in aerial photographs', in 'Proceedings of the South African IEEE Symposium on Communications and Signal Processing COMSIG 97', IEEE, Grahamstown, South Africa, pp. pp 75–80.

[77] Baltsavias, E., Mason, S. & Stallmann, D. (1995), 'Use of DTMs/DSMs and orthoimages to support building extraction', Technical report, Institute of

Geodesy and Photogrammetry, Swiss Federal Institute of Technology (ETH), Zurich, Switzerland.

[78] Weidner, U. & Förstner, W. (1995), 'Towards automatic building extraction from high resolution digital elevation models', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 50, no. 4, pp. 38–49.

[79] Baltsavias, E.P. & Mason, S.O. (1997), 'Automated shack reconstruction using integration of cues in object space', *International Archives of Photogrammetry and Remote Sensing*, vol. 32, no. Part 3-4W2.

[80] Rodriguez, C.E., Harwood, D. & Davis, L.S. (1994), 'An appearance-based approach to object recognition in aerial images', Technical Report CS-TR-3380, Computer Vision Laboratory: Center for Automation Research, University of Maryland.

[81] Kass, M., Witkin, A. & Terzopoulos, D. (1988), 'Snakes: Active contour models', *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331.

[82] Henricsson, O. (1998), 'The role of color attributes and similarity grouping in 3-D building reconstruction', *Computer Vision and Image Understanding*, vol. 72, no. 2, pp. 163–184.

[83] Mason, S., Rüther, H. & Smit, J. (1997), 'Investigation of the Kodak DCS460 digital camera for small-area mapping', *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 52, no. 5, pp. 202–214.

[84] Mayunga, S.D., Coleman, D.J. & Zhang, Y. (2008), 'Semi-automatic system for building extraction in dense urban settlement areas from high-resolution satellite imagery', URL `http://citeseerx.ist.psu.edu/viewdoc/summary10.1.1.112.4409`.

[85] Mayunga, S.D., Coleman, D.J. & Zhang, Y. (2005), 'Semi-automatic building extraction utilizing QuickBird imagery', in U. Stilla, F. Rottensteiner & S. Hinz (editors), 'Proceedings of the ISPRS Workshop CMRT 2005, 36(Part 3 / W24)', pp. 131–136.

[86] Žunić, J. & Rosin, P.L. (2003), 'Rectilinearity measurements for polygons', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 9, pp. 1193–1200.

[87] Mayer, S. (2001), 'Constrained optimization of building contours from high-resolution orth-images', in 'Proceedings of the International Conference on Image Processing', IEEE.

[88] Definiens (2007), *Developer 7 User Guide*, Definiens AG, Trappentreustr. 1, D-80339 München, Germany, 7.0.0.828 edn.

[89] Mayer, H. (1996), 'Abstraction and scale-space events in image understanding', *International Archives of Photogrammetry and Remote Sensing*, vol. 31, no. B3/III, pp. 523–528.

[90] Lindeberg, T. & ter Haar Romeny, B.M. (1994), *Linear Scale-Space*, Kluwer Academic Publishers, Dordrecht, Netherlands, Series in Mathematical Imaging and Vision, pp. 1–79.

[91] Mallat, S.G. (1989), 'A theory for multiresolution signal decomposition: The wavelet representation', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693.

[92] Levitt, S. & Aghdasi, F. (1998), 'An investigation into the use of wavelets and scaling for the extraction of buildings in aerial images', in 'Proceedings of the South African IEEE Symposium on Communications and Signal Processing (COMSIG 98)', IEEE, pp. 133–138.

[93] Lofy, B. & Sklansky, J. (2001), 'Segmenting multisensor aerial images in class-scale space', *Pattern Recognition*, vol. 34, pp. 1825–1893.

[94] Gerke, M., Straub, B.M. & Koch, A. (2001), 'Building extraction from aerial imagery using a generic scene model and invariant geometric moments', in 'Proceedings of the IEEE/ISPRS joint Workshop on Remote Sensing and Data Fusion over Urban Areas', pp. 85–89.

[95] Escoda, O.D., Petrovic, A. & Vandergheynst, P. (2004), 'Multiresolution segmentation of natural images: From linear to non-linear scale-space representation', *IEEE Trans. Image Processing*, vol. 13, no. 8, pp. 1104–1114.

[96] ter Haar Romeny, B.M. (1996), 'Introduction to scale-space theory: Multiscale geometric image analysis', Technical Report No. ICU-96-21, Utrecht University, The Netherlands.

[97] Weickert, J. (1997), *A Review of Nonlinear Diffusion Filtering*, Springer, Berlin, *Lecture Notes in Computer Science*, vol. 1252, pp. 3–28.

[98] Niessen, W.J., Vincken, K.L., Weickert, J.A. & Viergever, M.A. (1997), 'Nonlinear multiscale representations for image segmentation', *Computer Vision and Image Understanding*, vol. 66, no. 2, pp. 233–245.

[99] Bosworth, J.H. & Acton, S.T. (2003), 'Morphological scale-space in image processing', *Digital Signal Processing*, vol. 13, no. 2, pp. 338–367.

[100] Koenderink, J.J. (1984), 'The structure of images', *Biological Cybernetics*, vol. 50, pp. 363–370.

[101] Platel, B., Florack, L.M.J., Kanters, F.M.W. & ter Haar Romeny, B.M. (2003), 'Multiscale hierarchical segmentation', in 'Proceedings of the 9th ASCI Conference', Heijen, The Netherlands.

[102] Koster, A.S.E., Vincken, K.L., Graaf, C.N.D., Zander, O.C. & Viergever, M.A. (1997), 'Heuristic linking models in multiscale image segmentation', *Computer Vision and Image Understanding*, vol. 65, no. 3, pp. 382–402.

[103] Petrovic, A., Escoda, O.D. & Vandergheynst, P. (2004), 'Multiresolution segmentation of natural images: From linear to nonlinear scale-space representations', *IEEE Trans. Image Processing*, vol. 13, no. 8, pp. 1104–1114.

[104] Acton, S.T. & Mukherjee, D.P. (2000), 'Scale space classification using area morphology', *IEEE Trans. Image Processing*, vol. 9, no. 4, pp. 623–635.

[105] Lindeberg, T. (1998), 'Feature detection with automatic scale selection', *International Journal of Computer Vision*, vol. 30, no. 2, pp. 79–116.

[106] Lindeberg, T. (1993), 'Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention', *International Journal of Computer Vision*, vol. 11, no. 3, pp. 283–318.

[107] Stassopoulou, A., Caelli, T. & Ramirez, R. (2000), 'Automatic extraction of building statistics from digital orthophotos', *International Journal of Geographical Information Science*, vol. 14, no. 8, pp. 795–814.

[108] McCane, B. & Caelli, T. (1997), 'Multi-scale adaptive segmentation using edge and region-based attributes', in L.C. Jain (editor), 'Proceedings of the First International Conference of Knowledge-Based Intelligent Electronic Systems (KES '97)', IEEE Press, pp. 72–81.

[109] Shyu, C.R., Scott, G., Klaric, M., Davis, C.H. & Palaniappan, K. (2005), 'Automatic object extraction from full differential morphological profile in urban imagery for efficient object indexing and retrievals', in 'Proceedings of the ISPRS Joint Conference: URBAN 2005 and URS 2005', Tempe, USA.

[110] Brunn, A., Weidner, U. & Förstner, W. (1995), 'Model-based 2D-shape recovery', in G.S. et al (editor), 'Proceedings of the 17th DAGM Conference on Pattern Recognition (Mustererkennung)', Springer-Verlag, Berlin, pp. 260–268.

[111] Caelli, T., Stassopoulou, A. & Ramirez, R. (1998), 'Adaptive, multi-scaled variance models for image segmentation and feature extraction', in 'Proceedings of ISPRS, Commission III "Theory and Algorithms", 32(3/1)', pp. 202–206.

[112] Gerke, M., Straub, B.M. & Koch, A. (2001), 'Automatric detection of buildings and trees from aerial imagery using different levels of abstraction', *Publications of the German Society for Photogrammetry and Remote Sensing*, vol. 10, pp. 273–280.

[113] Ramer, U. (1972), 'An iterative procedure for polygonal approximation of plane curves', *Computer Graphics and Image Processing*, vol. 1, pp. 244–256.

[114] Douglas, D.H. & Peucker, T.K. (1973), 'Algorithm for the reduction of the number of points required to represent a line or its caricature', *The Canadian Cartographer*, vol. 10, no. 2, pp. 112–122.

[115] Levitt, S.P. & Dwolatzky, B. (1999), 'BuRS: A building recognition system', *South African Computer Journal, Special Issue: SAICSIT '99*, , no. 24, pp. 68–76.

[116] Suetens, P., Fua, P. & Hanson, A.J. (1992), 'Computational strategies for object recognition', *ACM Computing Surveys*, vol. 24, no. 1, pp. 5–61.

[117] Perona, P. & Malik, J. (1990), 'Scale-space and edge detection using anisotropic diffusion', *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639.

[118] Black, M.J., Sapiro, G., Marimont, D. & Heeger, D. (1988), 'Robust anisotropic diffusion', *IEEE Trans. Image Processing*, vol. 7, no. 3, pp. 421–432.

[119] Smolka, B. & Lukac, R. (2002), 'On the combined forward and backward anisotropic diffusion scheme for the multispectral image enhancement', in 'Proceedings of the ISPRS Commission III Symposium: "Photogrammetric Computer Vision", part B', pp. 249–254.

[120] Guichard, F., Moisan, L. & Morel, J.M. (2002), 'A review of P.D.E. models in image processing and image analysis', *Journal de Physique IV*, vol. 12, pp. 137–154.

[121] Weickert, J. & Benhamouda, B. (1997), 'Why the Perona-Malik filter works', Technical Report DIKU-TR-97/22, Department of Computer Science, University of Copenhagen.

[122] Rosin, P.L. & Žunić, J. (2005), 'Measuring rectilinearity', *Computer Vision and Image Understanding*, vol. 99, pp. 175–188.

[123] Kolesnikov, A. & Fränti, P. (2003), 'Polygonal approximation of closed contours', in G. Goos, J. Hartmanis & J. van Leeuwen (editors), 'Proceedings of the Scandinavian Conference on Image Analysis-SCIA2003', *Lecture Notes in Computer Science*, vol. 2749, pp. 778–785.

[124] Kolesnikov, A. (2003), *Efficient Algorithms for Vectorization and Polygonal Approximation*, Ph.D. thesis, University of Joensuu, Computer Science, Joensuu, Finland.

[125] Latecki, L.J. & Rosenfeld, A. (2002), 'Recovering a polygon from noisy data', *Computer Vision and Image Understanding*, vol. 86, pp. 32–51.

[126] Latecki, L.J. & Lakämper, R. (1999), 'Convexity rule for shape decomposition based on discrete contour evolution', *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 441–454.

[127] Latecki, L.J. & Lakämper, R. (1999), 'Polygon evolution by vertex deletion', in 'Proceedings of the Second International Conference on Scale-Space Theories in Computer Vision', Springer-Verlag, *Lecture Notes In Computer Science*, vol. 1682, pp. 398–409.

[128] Huertas, A., Lin, C. & Nevatia, R. (1993), 'Detection of buildings from monocular views of aerial scenes using perceptual organization and shadows', in 'Proceedings of the 1993 ARPA Image Understanding Workshop', Washington D.C.

[129] Tizhoosh, H.R. (1998), 'Fuzzy image processing: Potentials and state of the art', in 'Proceedings of the 5th International Conference on Soft Computing', vol. 1, pp. 321–324.

[130] Mees, W. & Acheroy, M. (1995), 'Automated interpretation of aerial photographs using local and global rules', in 'Proceedings of ANNIE'95: Conference on Artificial Neural Networks In Engineering', St. Louis, Missouri, USA.

[131] Mees, W. (1995), 'Image interpretation using fuzzy expert systems', in 'Proceedings of the International Conference on Digital Signal Processing (DSP95), Limassol, Cyprus, June', pp. 511–518.

[132] Kang, H.B. & Walker, E.L. (1994), 'Multilevel grouping: Combining bottom-up and top-down reasoning for object recognition', in 'Proceedings of the 12th IAPR International Conference on Pattern Recognition', vol. 1, pp. 559–562.

[133] The MathWorks (2002), 'Fuzzy logic toolbox user guide', Fuzzy Logic Toolbox, version 2.1.2, for MATLAB, version 6.5, release 13.

[134] Cox, E. (1999), *The Fuzzy Systems Handbook*, Academic Press, San Diego, second edn.

[135] Xie, M., Fu, K. & Wu, Y. (2006), 'Building recognition and reconstruction from aerial imagery and LIDAR data', in 'Proceedings of the International Conference on Image Processing', IEEE.

[136] Jain, R., Kasturi, R. & Schunk, B.G. (1995), *Machine Vision*, McGraw-Hill, New York.

[137] The MathWorks (Date last accessed: 08/04/2009), 'MATLAB and Simulink for technical computing', `http://www.mathworks.com/`.

[138] Informal Settlement Project (Date last accessed: 08/04/2009), 'Spatial information technologies to support urban planning in informal settlements', `http://www.sli.unimelb.edu.au/informal/`.

Appendix A

Detailed Results for Test Images

The detailed results for each of the images tested are presented here. For each image the following is given:

- the source image,
- the source image, with ground truth and system hypotheses superimposed,
- a schematic of the ground truth with overlaid system hypotheses,
- performance metrics based on buildings counts, area and shape accuracy,
- the image-specific parameters used,
- a histogram illustrating the frequency with which stack levels give rise to hypotheses appearing in the final interpretation of the scene,
- performance metrics by detection stage, and for different types of boundaries.

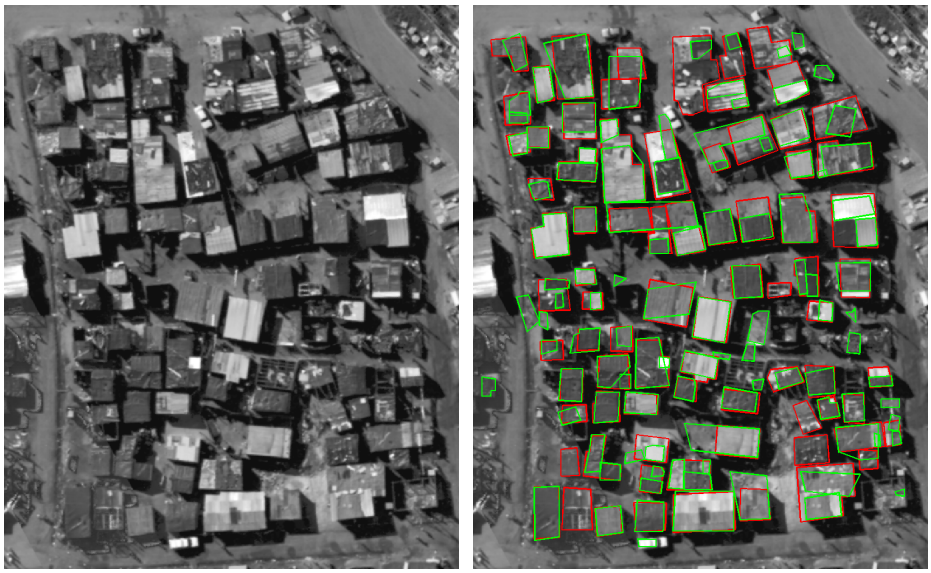
Note that for the 'Marconi Beam 1' image only the image-specific parameters are given, the rest of the results can be found in Chapter 6.

A.1 Marconi Beam 1

Image-Specific Parameters	
Image Size (pixels)	572×569
Ground Pixel Resolution (m)	0.18
Area Range (pixels^2)	[549 2737]
Sun Vector (length; angle)	17.4; $-71.0°$
Shadow Threshold	60

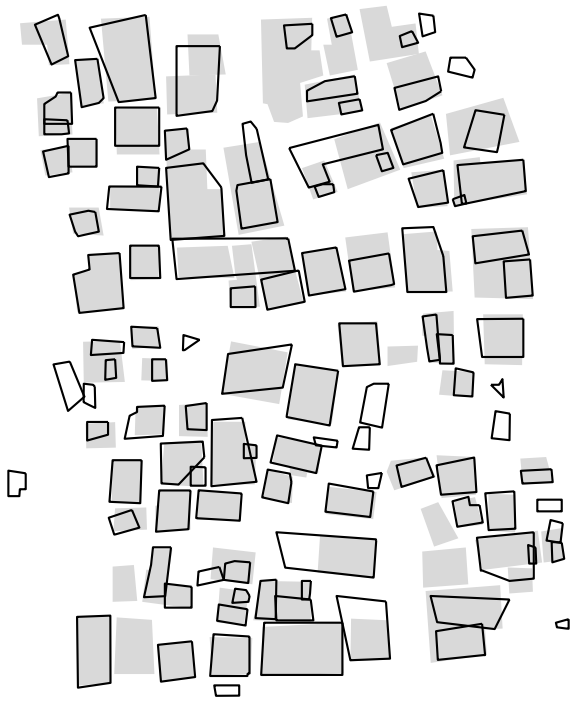
Table A.1: Image-specific parameters for the 'Marconi Beam 1' image.

A.2 Marconi Beam 2



(a) Source image.

(b) Ground truth polygons (in red) and system hypotheses (in green) overlaid on the source image.



(c) Schematic of ground truth (shaded areas) overlaid with system hypotheses.

Figure A.1: Results for the 'Marconi Beam 2' image.

Building-Count Metrics		Area-Based Metrics	
Buildings Detected (TP)	72	Branching Factor	0.15
Buildings Missed (FN)	5	Miss Factor	0.42
Non-buildings Detected (FP)	14	Detection Percentage	70.35
Detection Percentage	93.51	Quality Percentage	63.44
Quality Percentage	79.12		

Shape Accuracy Metrics		Image-Specific Parameters	
DP Mean Accuracy	74.43	Image Size (pixels)	813 × 1008
QP Mean Accuracy	64.64	Ground Pixel Resolution (m)	0.18
Mean Accuracy (Equation 6.1)	60.81	Area Range (pixels^2)	[495 3572]
		Sun Vector (length; angle)	12.5; −40.4°
		Shadow Threshold	55

Table A.2: All metrics for the 'Marconi Beam 2' image.

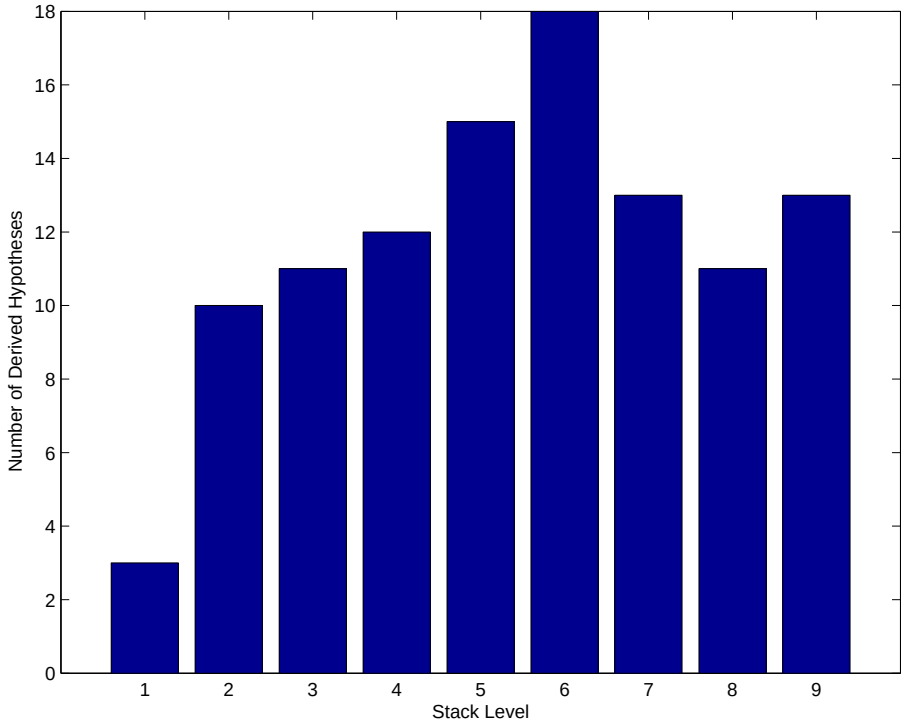
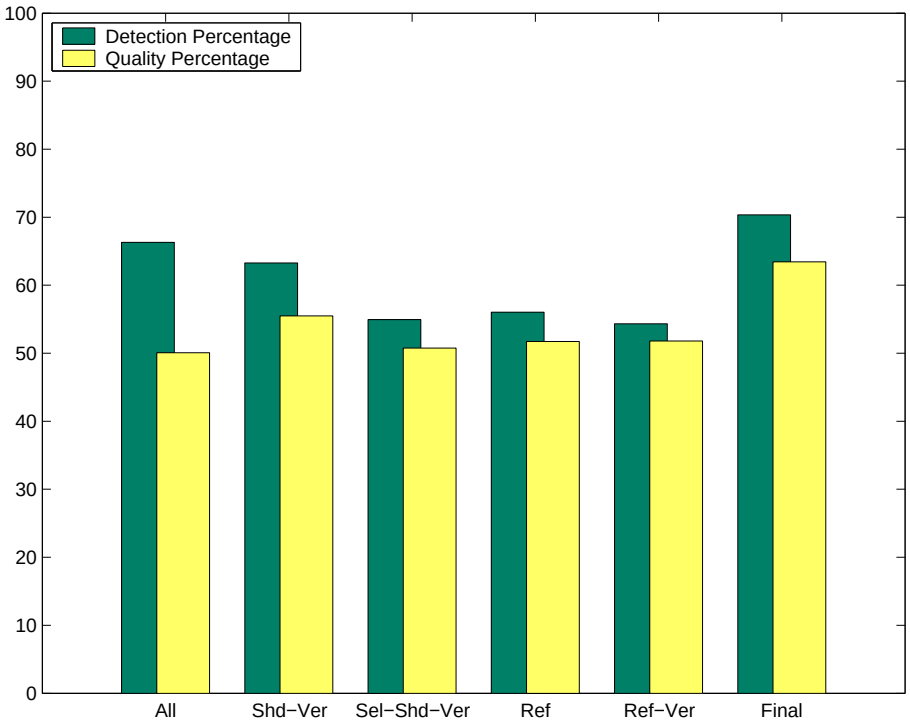
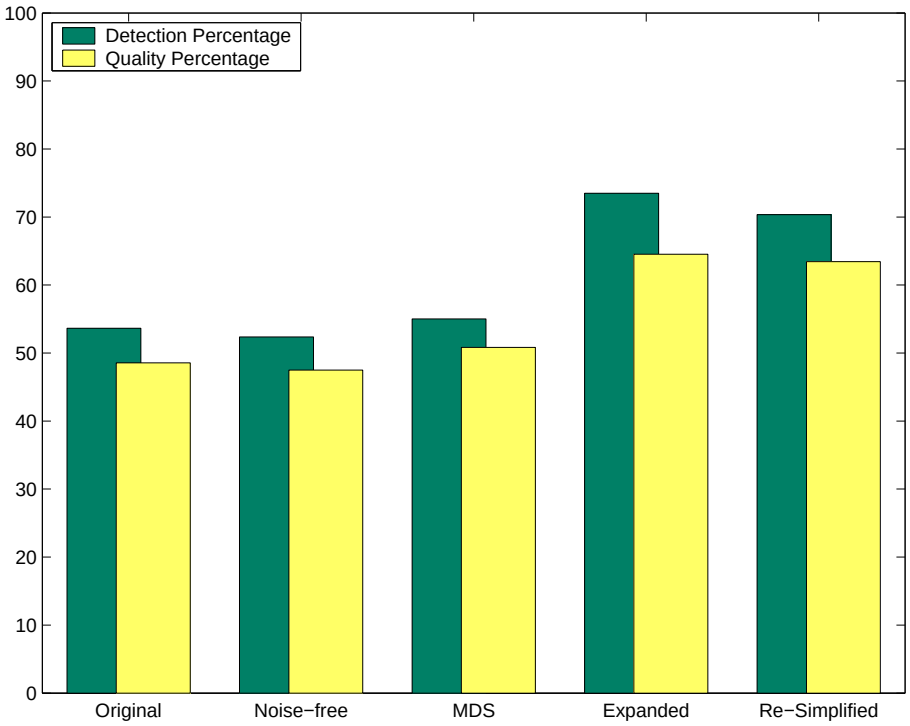


Figure A.2: Histogram of stack levels from which verified hypotheses are derived for the 'Marconi Beam 2' image.



(a) Metrics at different stages of the detection process. Boundaries resulting from model-driven simplification are used wherever possible.

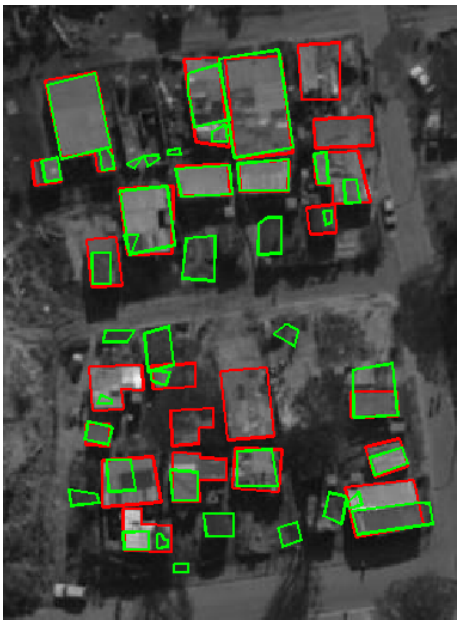


(b) Metrics for verified hypotheses with different types of boundaries

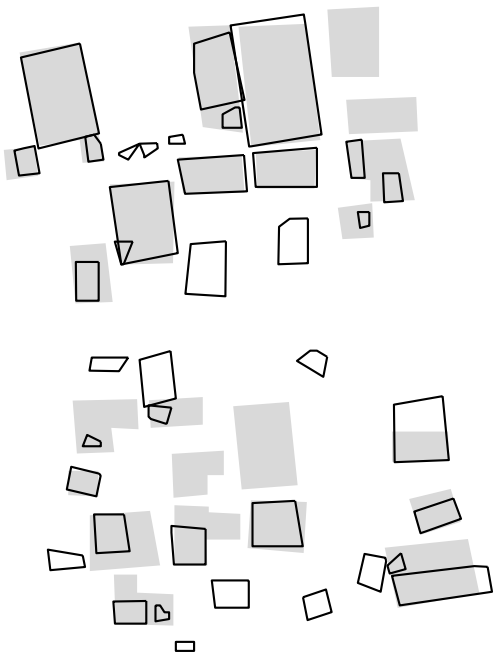
Figure A.3: Area-based performance metrics (detection and quality percentages) for the 'Marconi Beam 2' image at different stages of the detection process and for different boundary types.

A.3 Imizamo Yethu

(a) Source image.



(b) Ground truth polygons (in red) and system hypotheses (in green) overlaid on the source image.



(c) Schematic of ground truth (shaded areas) overlaid with system hypotheses.

Figure A.4: Results for the 'Imizamo Yethu' image.

Building-Count Metrics		Area-Based Metrics	
Buildings Detected (TP)	20	Branching Factor	0.33
Buildings Missed (FN)	4	Miss Factor	0.83
Non-buildings Detected (FP)	12	Detection Percentage	54.64
Detection Percentage	83.33	Quality Percentage	46.33
Quality Percentage	55.56		

Shape Accuracy Metrics		Image-Specific Parameters	
DP Mean Accuracy	59.61	Image Size (pixels)	380×514
QP Mean Accuracy	54.49	Ground Pixel Resolution (m)	≈ 0.18
Mean Accuracy (Equation 6.1)	59.66	Area Range (pixels^2)	[202 1682]
		Sun Vector (length; angle)	16.0; $-95.5°$
		Shadow Threshold	35

Table A.3: All metrics for the 'Imizamo Yethu' image.

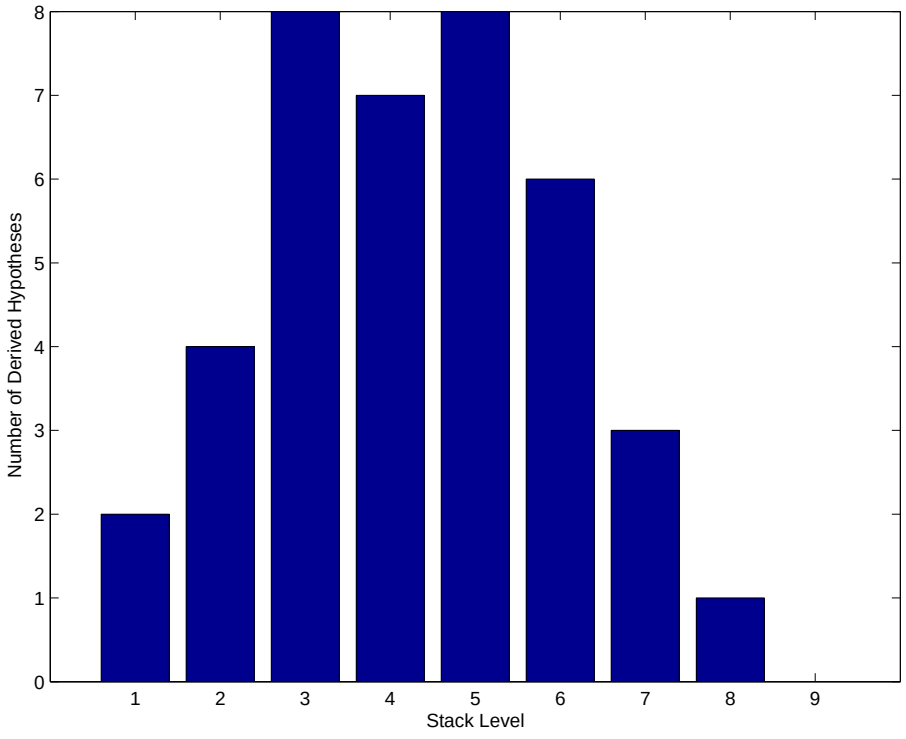
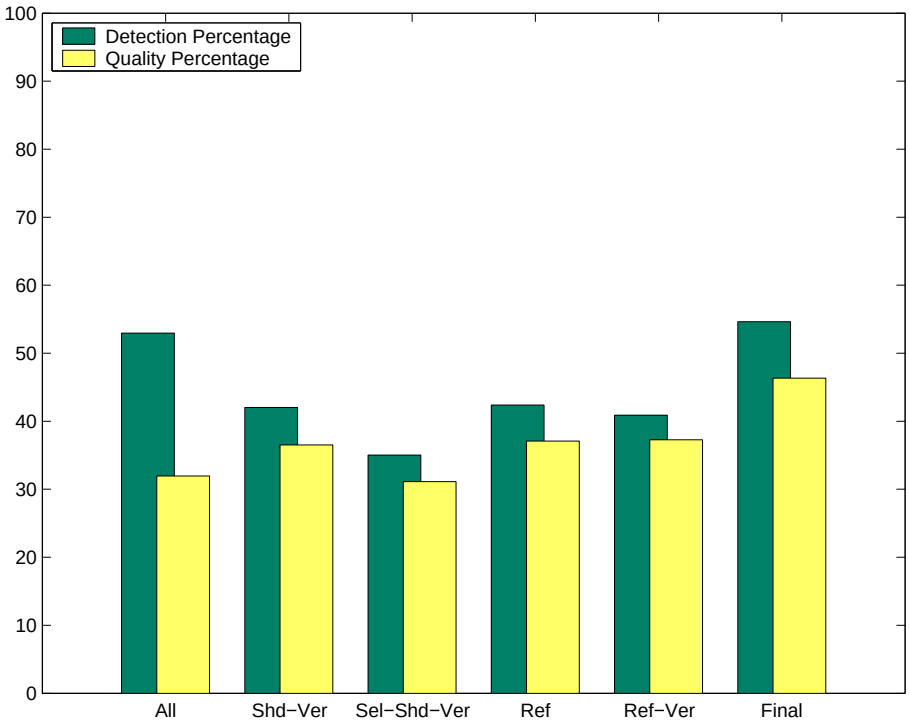
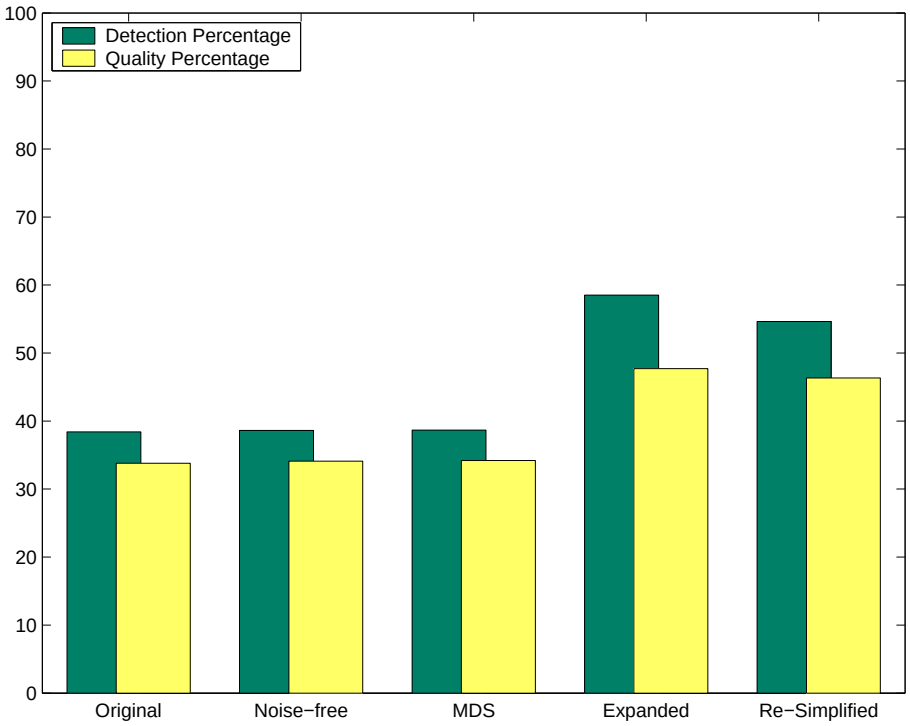


Figure A.5: Histogram of stack levels from which verified hypotheses are derived for the 'Imizamo Yethu' image.



(a) Metrics at different stages of the detection process. Boundaries resulting from model-driven simplification are used wherever possible.



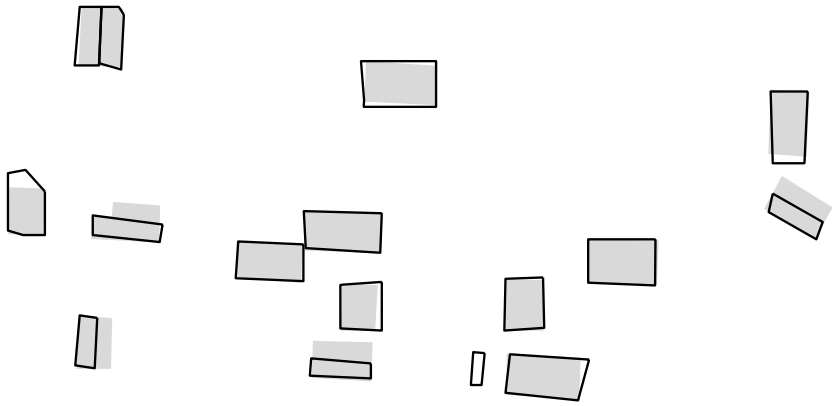
(b) Metrics for verified hypotheses with different types of boundaries

Figure A.6: Area-based performance metrics (detection and quality percentages) for the 'Imizamo Yethu' image at different stages of the detection process and for different boundary types.

A.4 Sparse Rural

(a) Source image.

(b) Ground truth polygons (in red) and system hypotheses (in green) overlaid on the source image.



(c) Schematic of ground truth (shaded areas) overlaid with system hypotheses.

Figure A.7: Results for the 'Sparse Rural' image.

Building-Count Metrics		Area-Based Metrics	
Buildings Detected (TP)	14	Branching Factor	0.10
Buildings Missed (FN)	0	Miss Factor	0.19
Non-buildings Detected (FP)	1	Detection Percentage	83.79
Detection Percentage	100.00	Quality Percentage	77.17
Quality Percentage	93.33		

Shape Accuracy Metrics		Image-Specific Parameters	
DP Mean Accuracy	82.61	Image Size (pixels)	751×537
QP Mean Accuracy	75.95	Ground Pixel Resolution (m)	≈ 0.3
Mean Accuracy (Equation 6.1)	79.38	Area Range (pixels^2)	[398 639]
		Sun Vector (length; angle)	7.9; $-142.8°$
		Shadow Threshold	35

Table A.4: All metrics for the 'Sparse Rural' image.

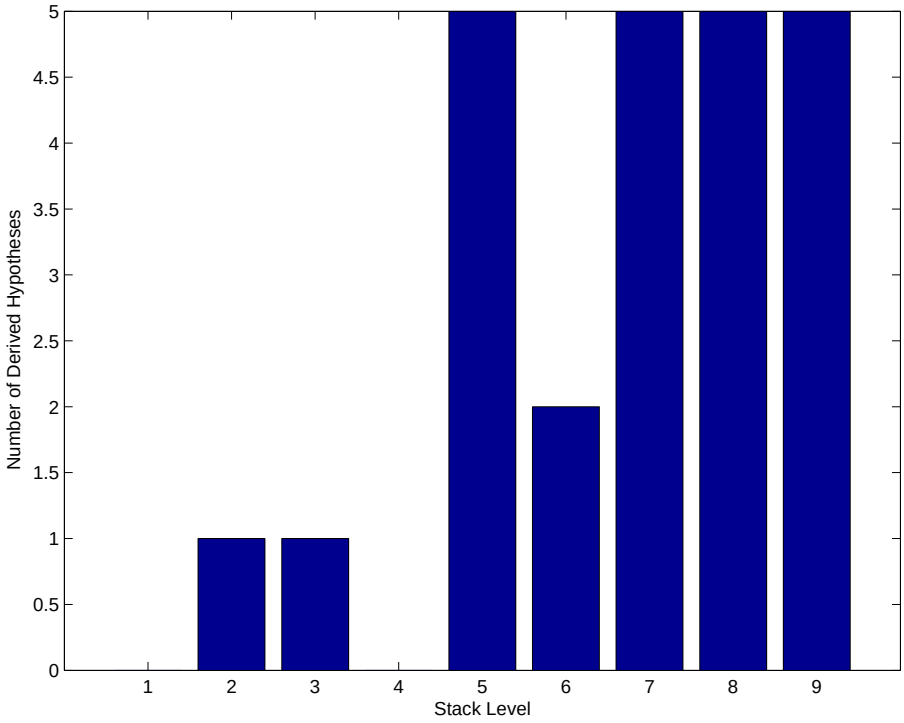
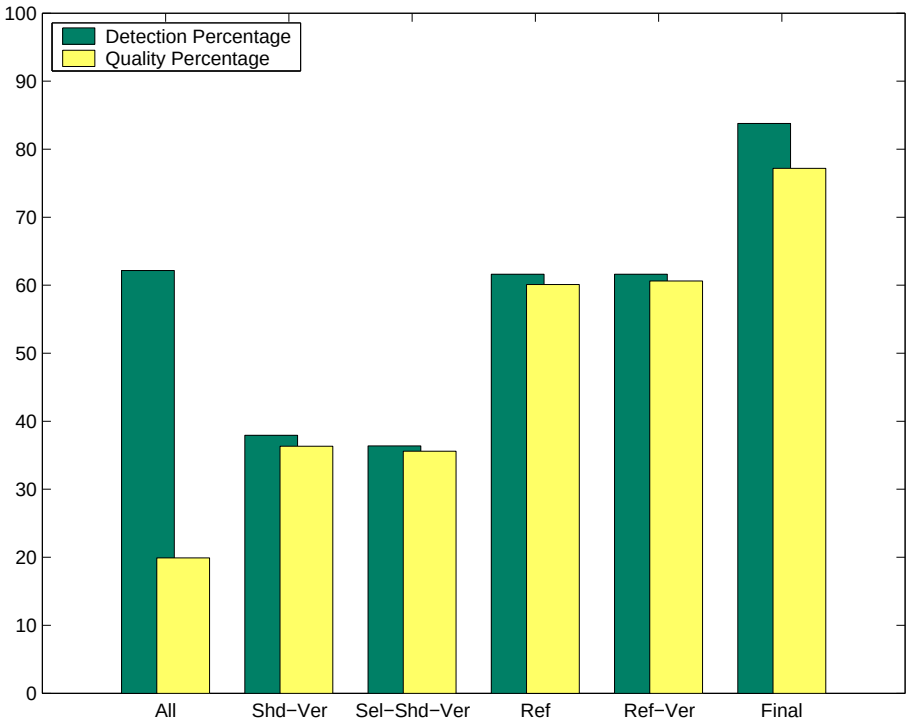
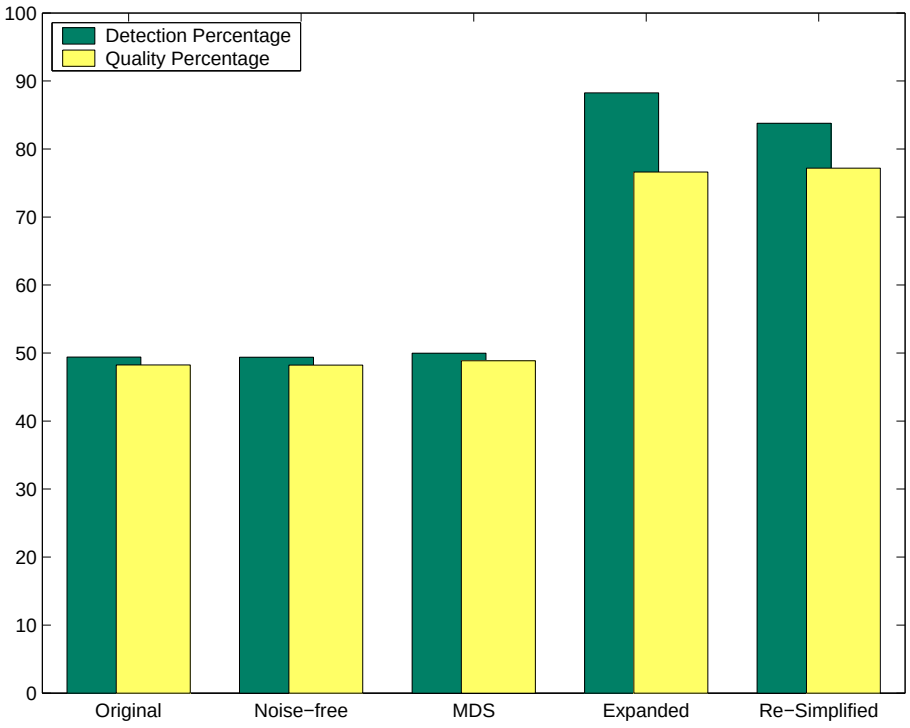


Figure A.8: Histogram of stack levels from which verified hypotheses are derived for the 'Sparse Rural' image.



(a) Metrics at different stages of the detection process. Boundaries resulting from model-driven simplification are used wherever possible.



(b) Metrics for verified hypotheses with different types of boundaries

Figure A.9: Area-based performance metrics (detection and quality percentages) for the 'Sparse Rural' image at different stages of the detection process and for different boundary types.

Appendix B

Image-Invariant Parameters

Table B.1 presents the constant system parameters. They are described briefly below. As explained in Section 6.7, these parameters work well across a variety of images, and have not been optimised for any particular image. A more rigourous approach for determining these parameters would be to employ machine learning techniques and to train the learning algorithm on a larger dataset.

The first two parameters both apply to the anisotropic diffusion model. The diffusivity constant, K, appears in Equation 5.3 while the rate of diffusion, λ, appears in Equation 5.5. Shadows are exaggerated through dilation with a disc-shaped structuring element having the radius given. The homogeneous operator (Section 5.2.4) is applied using a 3-by-3 window (HomogeneousOperatorWindowSize) with the homogeneity threshold as shown. The ShadowOverlapThreshold specifies by how much area (in percent) a region must overlap with identified shadow for the region to be rejected.

BoundarySamplesPerUnitLength is the number of samples to acquire per unit length of the roof-shadow boundary, while SunVectorSamples is the number of samples to acquire along the length of the specified sun vector. All samples are equidistantly spaced.

ShadowSupportThreshold, HypothesisSupportThreshold, and CombinedSupport-Threshold specify the support thresholds used in hypothesis verification (Section 5.7.1). MinimumSupportPercentage is the minimum percentage of samples that need to be contained by a candidate supporting hypothesis. Hypotheses which contain less than this number of samples are not considered as offering support.

Edges are determined using the Canny edge detector with the following parameters: CannyLowerThreshold, CannyUpperThreshold, and CannySigma. Edge

System Component	Parameter	Value
Anisotropic Diffusion	K (diffusivity constant)	15
Anisotropic Diffusion	λ (rate of diffusion)	0.25
Shadow Dilation	Disc-shaped Structuring Element Radius	1 pixel
Region Extraction	HomogeneousOperatorWindowSize	3×3 pixels
Region Extraction	HomogeneousOperatorHomogeneityThreshold	>3
Region Extraction	ShadowOverlapThreshold	>15%
Hypothesis Verification	BoundarySamplesPerUnitLength	1
Hypothesis Verification	SunVectorSamples	10
Hypothesis Verification	ShadowSupportThreshold	>0.3
Hypothesis Verification	HypothesisSupportThreshold	>0.3
Hypothesis Verification	CombinedSupportThreshold	>0.5
Hypothesis Verification	MinimumSupportPercentage	10%
Canny Edge Detection	CannyLowerThreshold	0.05
Canny Edge Detection	CannyUpperThreshold	0.1
Canny Edge Detection	CannySigma	0.2
Edge Approximation	MinimumEdgeLength	4 pixels
Edge Approximation	LineSegmentMaxDeviation	1 pixel
Boundary Expansion	EdgeWindowSearchBoundarySize	5
Boundary Expansion	EdgeAngleTolerance	30°
Boundary Expansion	DistanceFactor	1.5
Model-Driven Simplification	CompactnessWeight	1
Model-Driven Simplification	RectilinearityWeight	1
Model-Driven Simplification	PointSupportWeight	1
Model-Driven Simplification	ModelRotationThreshold	<15°
Model-Driven Simplification	MBShadowOverlapThreshold	<10%
Edge-based Verification	EdgeSupportAngleTolerance	20°
Edge-based Verification	EdgeSupportVerificationThreshold	>0.8
Grouping	GroupingRectilinearyFactor	0.75
Performance Analysis	BorderWidth	8 pixels
Performance Analysis	GTBorderWidth	5 pixels
Performance Analysis	ShadowSearchWindowWithinBorder	>75%

Table B.1: Summary of image-invariant parameters.

pixels produced by the Canny detector are linked together and approximated by straight lines. The minimum size of an edge formed by linking edge pixels is given by MinimumEdgeLength. The maximum deviation of the straight line approximation from the edge pixels is LineSegmentMaxDeviation. Straight line edge approximations are associated with reference boundary segments. A search window is constructed around each reference boundary and edges aligned with the reference boundary are identified and used to calculate corner points (Section 5.8.2). The distance of the search window vertices from the corresponding reference boundary vertices along the bisectors of the reference boundary vertices is given by EdgeWindowSearchBoundarySize. EdgeAngleTolerance specifies the maximum angle (in degrees) that is allowed between a boundary segment and its associated edge line. New corner points are generated by intersecting associated edge lines from adjacent boundary segments. These intersection points are excluded from the expanded boundary if they lie beyond a certain radius from the reference boundary centroid. This radius is determined by multiplying the distance of the furthest point on the reference boundary from the centroid by the factor, DistanceFactor.

Model-driven simplification takes places twice in the detection strategy presented – hypothesis boundaries are initially simplified using only rectilinearity and compactness (Section 5.5), and they are re-simplified after expansion where point support comes into play (Section 5.8.3). The weightings used are specified by CompactnessWeight, RectilinearityWeight, and PointSupportWeight. During model-driven simplification the amount of rotation of the canonical axis is restricted (see the rotation threshold, R, in Algorithm 5.1). ModelRotationThreshold gives the rotation threshold in degrees. Boundaries produced by model-driven simplification may overlap with identified shadow. The amount of shadow overlap that is tolerated for a model-driven-simplified boundary is specified by MBShadowOverlapThreshold.

Verification of the reference boundaries is based on the presence of at least one surrounding edge, having a maximum angle specified by EdgeSupportAngleTolerance, with respect to a segment of the boundary, and with edge shadow support greater than EdgeSupportVerificationThreshold.

The decision as to whether to form a grouping of boundaries or not is based on the acceptance criteria given in Equation 5.18. The constant in this equation is termed the GroupingRectilinearyFactor.

Hypotheses and ground truth close to the borders of the image are excluded from performance analysis in order not to prejudice the results. BorderWidth defines

the width of a border (in pixels) internal to the image. This border has the same aspect ratio as the image border and any system hypotheses which extend beyond this border are excluded from analysis. A separate border width for the ground truth is used, GTBorderWidth. The ground truth border is slightly narrower. This is to prevent ground truth boundaries being excluded while the corresponding shack boundary is found because, in general, a system hypothesis is smaller in area than the corresponding ground truth. A ground truth polygon or system hypothesis is also excluded from performance analysis if a sufficient number of samples along its roof-shadow boundary are beyond the image border. All the roof-shadow boundary samples lie within a defined window and the area of this window, as a percentage, that is required to be within the image border is given by ShadowSearchWindowWithinBorder.

Appendix C

Dynamic Membership Functions and the Matlab FIS File

The parameters of the membership functions for the input variables *Size* and *Support* are determined dynamically based on attributes of the hypotheses extracted from the image being processed. These attributes are calculated for shadow-verified hypotheses which have undergone model-driven simplification. The parameters of the *Size* and *Support* membership functions are:

- The triangular *Size* membership function "Small"
  - Domain: [`smallest-size median-size`]
  - Peak: `smallest-size`
- The triangular *Size* membership function "Medium"
  - Domain: [`smallest-size largest-size`]
  - Peak: `median-size`
- The triangular *Size* membership function "Large"
  - Domain: [`mean-size largest-size`]
  - Peak: `largest-size`
- The Z-shaped *Support* membership function "Low"
  - Domain: [`smallest-support midpoint-support`]
- The S-shaped *Support* membership function "High"
  - Domain: [(`smallest-support` + 1/3*(`support-domain-width`)) `largest-support`]

All other membership functions and the fuzzy rule base are hard-coded and image-invariant.

The text below forms the contents of the ".fis" file that is generated by MATLAB's Fuzzy Logic Toolbox (version 2.1.2, for MATLAB, version 6.5, release 13). This file stores the structure and parameters for a single fuzzy inference system. The ".fis" file given here is that used for resolving conflicting hypotheses on the Marconi Beam image.

The rules matrix (at the end of the file) can be understood as follows: the first four columns represent the input variables, the fifth column represents the output variable, the sixth column is the rule weighting and the seventh column indicates the conjunction that is used in the antecedent (AND in all cases). For the input and output variables, the number in the column indexes the specific membership function involved in the rule.

```
[System]
Name='marconi__fis'
Type='mamdani'
Version=2.0
NumInputs=4
NumOutputs=1
NumRules=17
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='sum'
DefuzzMethod='centroid'

[Input1]
Name='Size'
Range=[36 2339]
NumMFs=3
MF1='Small':'trimf',[36 36 286]
MF2='Medium':'trimf',[36 286 2339]
MF3='Large':'trimf',[419.111344537815 2339 2339]

[Input2]
Name='Rectilinearity'
Range=[0 1]
NumMFs=3
MF1='Low':'zmf',[0 0.7]
```

```
MF2='Medium':'pimf',[0.2 0.5 0.5 0.8]
MF3='High':'smf',[0.3 1]

[Input3]
Name='Compactness'
Range=[0 1]
NumMFs=2
MF1='Medium':'trapmf',[0 0 0.785398163397448 1]
MF2='High':'trapmf',[0.785398163397448 1 1 1]

[Input4]
Name='Support'
Range=[0.3 1.79154929577465]
NumMFs=2
MF1='Low':'zmf',[0.3 1.04577464788732]
MF2='High':'smf',[0.797183098591549 1.79154929577465]

[Output1]
Name='Likelihood'
Range=[0 100]
NumMFs=5
MF1='VeryUnlikely':'zmf',[0 35]
MF2='Unlikely':'pimf',[0 25 25 50]
MF3='Maybe':'pimf',[25 50 50 75]
MF4='Likely':'pimf',[50 75 75 100]
MF5='VeryLikely':'smf',[75 100]

[Rules]
3 0 2 0, 5 (1) : 1
2 0 2 0, 3 (1) : 1
1 0 2 0, 3 (1) : 1
3 0 1 0, 1 (1) : 1
2 0 1 0, 3 (1) : 1
1 0 1 0, 2 (1) : 1
3 3 0 0, 4 (1) : 1
2 3 0 0, 3 (1) : 1
1 3 0 0, 3 (1) : 1
3 2 0 0, 3 (1) : 1
```

```
2 2 0 0, 3 (1) : 1
1 2 0 0, 3 (1) : 1
3 1 0 0, 1 (1) : 1
2 1 0 0, 2 (1) : 1
1 1 0 0, 2 (1) : 1
0 0 0 2, 1 (1) : 1
0 0 0 1, 3 (1) : 1
```