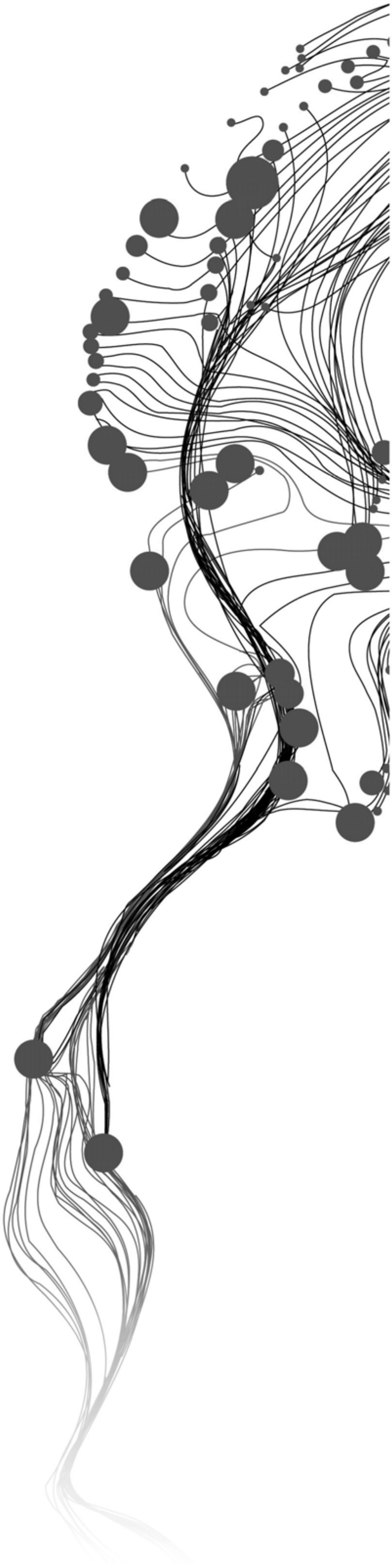
EXPLORATION OF SCALE-SPACE THEORY FOR MULTI-RESOLUTION SATELLITE IMAGE ANALYSIS

KIPYEGON BENARD LANGAT
February, 2011

SUPERVISORS:
Dr. V.A. Tolpekin
Dr. M. Gerke



EXPLORATION OF SCALE-SPACE THEORY FOR MULTI-RESOLUTION SATELLITE IMAGE ANALYSIS

KIPYEGON BENARD LANGAT
Enschede, The Netherlands, March, 2011

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization: Geoinformatics

SUPERVISORS:
Dr. V.A. Tolpekin
Dr. M. Gerke

THESIS ASSESSMENT BOARD:
Prof.Dr.Ir. A. Stein (Chair)
Ms Dr. A. Dilo (External Examiner)

ABSTRACT

Remotely sensed image representation is an important property in image processing for research and land cover mapping. It is natural that 'good' image representation leads to better quality of information derived from images. Availability of image data in multi-resolution representation determines the level of details and accuracies translated to maps. Map accuracy is further affected by the mixed pixels problem. Generalisation of maps from fine resolution images is a challenging process due to high level of details present in an image. A multi-resolution approach for image analysis is more desirable because it allows more flexibility in selection of image resolutions for particular interpretation. A multi-scale image analysis based on image structure for remotely sensed imagery is presented.

Scale-space theory is a multi-scale technique for analysing images across various scales. Based on scale-space theory, multi-scale representation of an image is computed using a Gaussian function of increasing width. Scale-space image analysis results in multi-resolution images that can be used as input for varying making maps of different scales. Scale-space image derivatives have been computed using derivatives of the Gaussian function up to the second order. Scale-space features are detected from the image derivatives and tracked with the coarsening of scale. An analysis of detected scale-space points based on image scene characteristics shows a relation in distribution of points to image heterogeneity. A linear reconstruction algorithm based on scale-space features has been presented. A Gaussian kernel based resampling is developed and its process is a weighted interpolation of new image values with more emphasis from neighbouring pixels than distant ones.

Both synthetic and satellite images have been used. Scale-space representation provides a hierarchical decomposition of satellite images structure that can easily be exploited compared to fixed resolution images. Scale-space points degenerates monotonically in scale-space with their distribution influenced by objects, edges, illumination and noise. The reconstruction algorithm presents interesting results based on random, equidistant and scale space points. A linear correlation between Gaussian kernel based resampling and scale-space representation is presented.

Key words: *Scale-space theory, scale parameter, reconstruction, Gaussian function, resampling, scale-space representation, image derivatives, scale-space points, critical points, saddle points, top points, blobs and maps*

ACKNOWLEDGEMENTS

The research presented in this document was done at University of Twente – Faculty of Geoinformation Science and Earth Observation (ITC) in the Netherlands as part of attainment of Msc Geoinformatics degree. Towards my achievement, I greatly appreciate ITC and Nuffic who made it possible for me to study through the Netherlands Fellowship Programme. To the staff of these organizations, thank you. I also express my appreciation to the assessment board committee of this thesis for taking time to read this document and making sense of out it and my propositions.

Next is my heartfelt appreciation to my first supervisor Dr. V.A. Tolpekin to whom I owe a lot for the completion of this work. Without his guidance and showing me the way out when it seemed impossible it could have been very difficult for me. I also appreciate Dr. M. Gerke, my second supervisor, for his contribution and feedback which helped me improve my research. My special thanks to Mr G. Huurnemann and Dr. Wietske Bijker who offered me a lot of cooperation when I served as representative for Geoinformatics class. I further appreciate the spirit of my friends Kim, Panday, John, Nazanin and Eriminah who gave valueless encouragement and company during the research period.

My sincere thanks go to Msc Geoinformatics class who not only offered me an opportunity to serve as their representative but also a lot of cooperation. I thank them all. I would not forget colleagues whom we served together in the Students Association Board for the good moments we had in serving the student community. A lot of thanks also go to the student affairs for the cordial relationship we had during my tenure. I would also like to thank Kenyan community for the togetherness we had. Special thanks to Nthiwa, Ngugi, Mathenge, Alando and Caro; you made me feel at home far away from home and I thank you all.

Finally, to my family; I am so grateful! Agnes and Mark you have persevered a lot due to my absence and your encouragement and prayers always kept me going. I dedicate all these work to you. I also thank mum, dad and our entire family.

To God, Thank you so much!

Thank you all!

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. Background and motivation

Remotely sensed imagery has gained a lot of credence in spatial data acquisition due to its effectiveness than the conventional ground surveys. Management of resources or emergency operations covering extensive areas require timely data acquisition methods, an option achieved by remote sensing. This preference of remotely sensed data has been supported by Jong et al. (2004) and Tso et al. (2001). The popularity of the use of remotely sensed imagery in environmental processes can be attributed to the numerous manipulations that can be applied to the data for specific applications. Furthermore, the sensor's synoptic view, high repeatability, multi-purpose and data acquisition in areas inaccessible by ground survey are some of the reasons attributed to its popularity.

Remote sensing data are processed into a knowledge base of land use and land cover maps used to support decision making in management and planning of geo-information activities. Map production is influenced by several factors mainly input data characteristics. The accuracy of the output map influences the quality of decisions made out of them. Accuracy of map production can be linked directly or indirectly to the trade-offs between the spatial, spectral and temporal resolution of the sensor (Hughes et al., 2010). This trade-off results in data that rarely fits a particular use. Since the inception of earth observation satellites, spatial, spectral, temporal and radiometric resolutions have been advancing in the quest to acquire data fit for particular uses. Spatial resolution is considered the most essential sensor property in image processing (Romeny, 2002).

In conventional land cover map making, spatial resolution of input image is propagated into the accuracy of the output map. Spatial resolution is the size of ground area from which the sensor receives the radiation which translates into the size of the smallest possible feature detected by the sensor. A sensor's spatial resolution is influenced by its instantaneous field of view (Tso et al., 2001). The interpretation of satellite images of varying spatial resolution in map production is complicated by presence of more than one class in the same pixel leading to maps which may not always be desirable to the user. These pixels containing more than one class are referred to as mixed pixels or mixels. This has lead to development of post processing techniques to enhance final map accuracy and quality. An understanding of satellite imagery for production of accurate maps has always been a problem due to the uncertainty of pixel composition (Fisher, 1997). The contention of this elementary unit of analysis has motivated researchers to develop image post processing techniques with an aim of reducing the uncertainty associated with it.

Estimation of the composition of mixed pixels in terms of land cover classes is an active research area for sub-pixel classification methods. Sub-pixel classification methods are being used with an aim of solving the contextual mixed pixel problem which increases with decrease in spatial resolution (Atkinson, 2004). These techniques address the hard classification associated with pixel classification methods where a pixel despite spatial and spectral similarity to more than one class is assigned to one class. In sub-pixel classification, only the proportion of classes within a pixel can be derived but the spatial distribution of the classes remain unknown (Jong et al., 2004). Super resolution mapping (SRM) is one of the creative land cover mapping techniques where the derived map has a finer spatial resolution than the input image (Atkinson et al., 1997). It is used to surpass the limitations of optical sensors through the use of algorithms by transforming the soft classification of mixed pixel into finer scale hard classified maps (Atkinson et al., 1997; Farsiu et al., 2004; Verhoeye et al., 2002)

1.2. Problem statement

Soft classification is a process that requires an in-depth understanding of image structure and associated image acquisition process. The circumstances on acquisition process have been advancing towards improvement of the spatial data resolution. In many circumstances manipulation of the sensor platform and sensor optics or imaging array is not always an easily available option (Akgun et al., 2004). The mixed pixel problem depending on the context is known to increase with decrease in spatial resolution. An understanding of the image structure in relation to the mixed pixel problem needs to be established. Lindeberg (1990) argued that a multi-scale image representation leads to systematic change of information content and hence provides image structure information by relating content at different resolution levels. The trend of image structure in scale-space would provide an additional insight to soft classification techniques.

One method of image structure analysis is through scale-space theory which involves image analysis across different scales. Scale-space features are believed to contain image information at varying levels of importance. Scale-space points are example of scale-space features believed to contain crucial image information (Kuijper et al., 2003; Nielsen et al., 2001). A scale-space point existing at certain level of scale can be tracked to a similar critical point both at slightly coarser and slightly finer scale (Lindeberg, 1994). Detection and tracking of these points is analysed to establish a relation of their presence with the image structure information over specific scales in a scale-space representation. The influence of noise in detection and tracking is not yet analysed. Kanters et al.(2003b) proposed an image minimum variance reconstruction technique using these points. The minimum variance reconstruction technique has been tested to work well with medical images (Kanters et al., 2003b) and it also appears potential for satellite images. In this method it is proposed that an image can be reconstructed from information at coarser scales. Up to date super resolution mapping techniques have not explored image structure for image reconstruction. Understanding the trend of image structure information in scale-space may provide an insight on reconstructing finer resolution image from coarse resolution input images.

1.3. Research objective and questions

The main objective of this research is to explore potential of scale-space theory for analysis of remotely sensed images

The following sub objectives are defined to reach the main objective

1. To detect scale-space points of satellite imagery and establish its relation with actual scene characteristics
2. Establish whether scale-space points are affected by presence of noise in images
3. Explore image reconstruction from coarse scale-space image information
4. Establish relationship between scale-space representation and kernel based resampling

1.4. Research questions

In the process of achieving the aforementioned objectives, the following questions are posed to guide the research process;

1. What kind of image objects contribute to scale-space points?
2. What can image structure scale-space points tell about actual scene characteristics?
3. How does noise influence tracking of points in scale-space representation?
4. How can the scale-space features be used for estimation of original image by reconstruction?
5. Does kernel based resampling have a relation with scale-space representation?

1.5. Overview of methodology

Figure 1.1 shows the general approach adopted in the execution of this study. The first stage involves data acquisition and pre-processing. After the first stage, a scale-space representation is computed through image convolution using Gaussian function of increasing width. A different approach of building scale-space representation using Gaussian kernel based resampling is also presented. An analysis of scale-space points' detection and tracking in relation to different scene characteristics is done. Finally a linear reconstruction approach by Kanters et al. (2003b) is used to analyse both synthetic and very high resolution images.

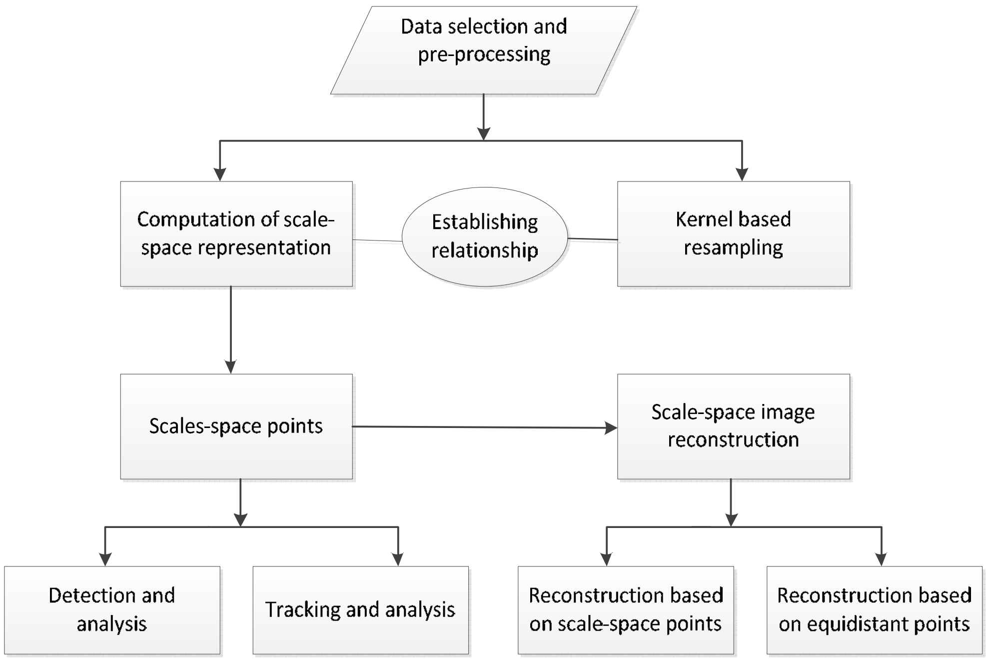


Figure 1.1: Methodology flowchart

As part of information extraction through image structure analysis, image reconstruction based on scale-space, random and equidistant points is assessed. The different reconstruction quality by the different types of points can partly inform on the image information contained in scale-space representation. Analysing the presence of scale-space features over its scale range could provide an insight on reconstruction of finer resolution image through tracking of scale-space points.

1.6. Outline of thesis presentation

This thesis report is presented in seven main chapters. Chapter one is the general introduction to the research which encompasses motivation, problem statement, stated objectives and research questions. Chapter two addresses the theory behind this research while Chapter three presents related applications. Data selection, data pre-processing and the adopted implementation approaches are explained in Chapter four. In chapter five, the results are presented and they are discussed in the context of the research objectives in Chapter 6. Finally chapter seven presents the conclusions and recommendations for further research.

2. THEORY

This chapter is presents the theory related to this work. The concepts of scale-space theory and definition of frequently used terms is explained. The mathematical concepts used in implementation are also presented.

2.1. Scale-space theory

Scale-space theory is a tool used for representation and analysis of image structure in various scales (Lindeberg, 1994). It is also referred to as deep structure. This type of multi scale representation is achieved by blurring an image using Gaussian function of increasing scale parameter. The output images have the same spatial sampling as the original image (Lindeberg, 1994; Romeny, 2002).

2.1.1. Introduction

In reality world objects possess natural property that they can be observed only over certain ranges of scale as 'meaningful entities' (Lindeberg, 1994). A famous example of multi scale representation is a tree crown used as an example by Linderberg (1994). and Romeny (2002). Scale-space concept of a tree crown is that it only 'makes sense' over a scale range of few centimetres to some few meters. This multi scale concept of a tree crown could be out of its scale range to be discussed at kilometre or micrometre level. Leaves of a tree in form of molecules are more relevant to be addressed at micrometer scale level while forest in which trees grow is more relevant to be addressed at kilometre scale level. Real world objects which naturally appear in different ways are best described depending on the scale of observation. Scale of observation is a well known concept in physics where phenomenon are modelled at several levels of scale (Lindeberg, 1998).

Human vision is also known to possess a multi scale visual ability well equipped for multi scale information extraction (Romeny, 2002). As an example, it is able to identify a building with windows, chimney, bricks as well as roof materials at the same time. Remote sensing devices integrate emitted or reflected radiation with digital camera charge-coupled device detector element. Different properties of these storage devices lead to multi-resolution remote acquisition of data and information. The appearance of landscape features varies in remotely sensed imagery of different spatial resolution (Heuwold et al., 2007). These sensor devices record the incoming energy as electrical signal in form of pixels whose size determines the resulting image sharpness.

Multi-scale image representation has in the recent past gained a lot of credence in computer vision, image processing and biological vision modelling (Lindeberg, 1994). This concept have commonly been used in the processing steps in large number of visual operations including feature detection, optic flow, and computation of shape cues (Lindeberg, 1998). According to Romeny (Romeny, 2002), scale is an essential parameter in computer vision research and it is of immediate importance for observation processes where measurements are administered. It brings about hierarchy, an important notion in image analysis (Romeny, 2002). The multi scale reality and human vision have been attributed as the inspirations behind scale-space theory. One of the crucial points in deep structure is that structures at coarse scales are generalizations of the corresponding finer scale structures (Kuijper et al., 2001; Lindeberg, 1994; Romeny, 2002; Weickert, 1998). The intention of suppression of structures with increase in scale parameter is to obtain separation of structures in the initial image (Lindeberg, 1998) by reduction of noise.

2.1.2. History and axioms of scale-space theory

Gaussian function is preferred for generating scale space representations because of a number of reasons. It is considered the most popular function in computer vision and gives a better normalization for factor for discrete and truncated versions. The function is qualitative in the sense that it is symmetric and it emphasizes neighbouring pixels than distant ones. This property reduces smoothening while maintaining noise averaging properties (Lindeberg, 1994). The quantitative Gaussian function properties include its smoothness and the possibility of differentiating infinitely many times. Additionally the Gaussian function is always positive and is separable. The set of blurred images with increasing scale parameter is what is referred to as (Gaussian) scale-space (Kuijper, 2002).

It is believed by Romeny (2002) and Kuijper (2002) that scale-space representation of images was pioneered by Iijima (1962). His work unfortunately remained unnoticed for a long time because it was published in Japanese. In western world the idea of scale-space was introduced by Witkin and Koenderink as depicted by Weickert, et al. (1999). Koenderink (1984) showed that the natural way to represent an image in infinite resolution is by convolution using Gaussian of various widths hence obtaining smoothened image at scale determined by the width.

From a multi scale perspective, a scale-space representation is a special type of multi scale comprising of continuous scale parameter and preservation of the same spatial sampling at all scales (Lindeberg, 1994; Stefanidis et al., 1993). Additionally, Gaussian kernel is singled out as a unique blurring kernel for describing the transformation from representation at finer to coarse scale representations (Lindeberg, 1994; Romeny, 2002; Weickert, 1998). There are various ways of deriving a continuous one-parameter family of signals from a given signal but Gaussian kernel is considered a special case. Among the special properties of this choice include; linearity, spatial shift invariance, isotropy and scale invariance. All these properties are combined with the notion that there should be no creation of structures in the transformation of an image from fine to coarser scales (Koenderink, 1984; Lindeberg, 1998). Gaussian generated scale-space also offers the advantage most notably when combining smoothing and edge detection (Stefanidis et al., 1993). Image edges are detected as discontinuities and thus correspond to image zero-crossings of a twice differentiated image. Combined with different ways of formalizing the notion, no new features should be created by the smoothing transformation (Florack et al., 1992; Lindeberg, 1994; Romeny, 2002). Gaussian derivatives are also used to analyse image grey value fluctuations in the neighbourhood for better understanding of image structure (Weickert, 1998).

2.2. Image

An image is a rectangular representation of a physical object in form of a grid containing pixel values. In this study, the physical object is part or whole of the earth's surface acquired by remote sensing. The pixel value is the smallest element of the grid with value corresponding to the radiation reflected from the earth's surface as detected by the sensor. The image is represented in form of columns and rows as shown in Figure 2.1 where they are mostly referred to as X and Y respectively. The different grey level depicted in the figure can be interpreted as group of pixels forming a feature of ground object. In this research the image if represented by a function f(x,y) where x and y are rows and columns. In image interpolation the image dimension has also referred to as image grid system.

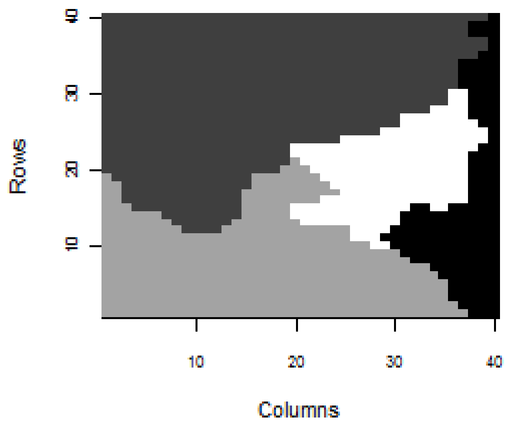


Figure 2.1Sample image representation

2.3. Gaussian function

$$G(\boldsymbol{x}, \boldsymbol{y}; t) = \frac{1}{4\pi t} e^{-\frac{x^2+y^2}{4t}} \tag{2.1}$$

This is a standard Gaussian function centred at zero, with a variance $2t$ and t is called the scale parameter in computation of scale-space representation. G is a two dimensional function with x as columns and y rows of the kernel. The selected scale parameters used to derive a scale-space representation of an image are called scale-space levels. The term $1/4\pi t$ is the normalization constant because the function's exponential integral is not unity. As the scale parameter increases the amplitude reduces substantially. The normalization ensures average grey-level invariance is achieved which means that the grey level of the image remains the same with kernel blurring.

2.4. Scale

This relates to image details which only exist as meaningful entities over limited scales. The presence of objects in scale-space representation depends on the scale of observation the derived image. Observation scale assumed to be the scale of data acquisition, spatial resolution. The extents of an image feature or object existing depends on its inner and outer scale (Lindeberg, 1994). The outer scale of an object is taken to correspond to the minimum size of the window that completely contains the object. Inner scale on the contrary is loosely taken to correspond to the scale at which sub-structures of an object begin to appear. Image features can only be defined within its scale range defined by its inner and outer scales. Resolution of images is determined by the scale of observation of the sensor. The observation scale directly influences the smallest size of the object that can be present in an image of particular resolution. Scale parameter as often used in this thesis refers to relates to varying of scale in an image from scale of observation, initial image scale, to a point when image features are considered decomposed. Scale parameter is related to the variance of the Gaussian kernel by the following formula;

$$t = 0.5\sigma^2 \tag{2.2}$$

Scale level is referred to selected discrete levels of scale along the scale dimension. Because image is a discrete object, scale parameter is selected at intervals from the possible scale parameter values.

2.5. Image convolution, derivatives and Laplacian

Computation of scale-space representation using convolution is as follows:

$$L(x, y; t) = G(x, y; t) \otimes f(x, y) = \iint G(x - x', y - y'; t)\ f(x', y')dx'dy' \tag{2.3}$$

Where;

t is the continuous scale parameter $t \geq 0$
$L(x, y; t)$ is the scale-space representation at pixel (x,y) and scale t
$f(x, y)$ is the original image
$\otimes$ denotes convolution

Image derivative is a new version of an image resulting from convolution by Gaussian function derivatives. In this research it used to refer to convolutions by first and second order Gaussian function derivatives. From Equation (2.3), it means that differentiation is made by integration leading to regularized derivatives of the input image. Image derivative with respect to x involves convolution of an image by differentiated Gaussian function with respect to x.

$$L_x(x, y; t) = \frac{\partial}{\partial x}(G(x, y; t)) \otimes f(x, y) \tag{2.4}$$

Image derivatives are achieved by convolution of the initial image with corresponding derivatives of a Gaussian function. The image derivatives are symbolised by; L_y, L_{xx}, L_{xy} and L_{yy}, computed by convolving f by $\partial_y G$, $\partial_{xx} G$, $\partial_x \partial_y G$ and $\partial_{yy} G$ respectively. Differentiation of an image up to any arbitrary order is achieved as follows (Kuijper et al., 2001);

$$\frac{\partial}{\partial x} L(x, y; t) = \frac{\partial}{\partial x}\left(G(x, y; t) \otimes f(x, y)\right) = \left(\frac{\partial}{\partial x} G(x, y; t)\right) \otimes f(x, y) \tag{2.5}$$

The same property holds for differentiation with respect to y. All the partial derivatives of the Gaussian kernel, in scale-space theory are solutions to the diffusion equation (Romeny, 2002) and together with the zero-th order Gaussian form complete family of scaled differential operators for this thesis. Consequently, in terms of differential equations, changes of image features in scale-space satisfy the diffusion equation as follows;

$$\frac{\partial L(x,y; t)}{\partial t} = \frac{\partial^2 L(x,y; t)}{\partial x^2} + \frac{\partial^2 L(x,y; t)}{\partial y^2} = \Delta L \tag{2.6}$$

Where $\Delta L(x, y; t)$ is the Laplacian, it is equal to the sum of the second order partial derivatives of an image. The diffusion equation analogy holds for scale-space image representation in that at "infinite" scale parameter, the derived image will have a constant value (Kuijper, 2002; Lindeberg, 1994; Romeny, 2002). The diffusion equation satisfies the maximum principle which Koenderink (1984) argues it is one of the reasons behind generation of scale-space theory. It states that the amplitude of local maxima always decrease with increase in scale parameter. Figure 2.2 is grey level illustration of the 2D discrete derivative approximation of Gaussian kernels up to the second order with scale parameter $t = 20$. The profiles in Figure 2.3 are derived from Figure 2.2 at different orientations.

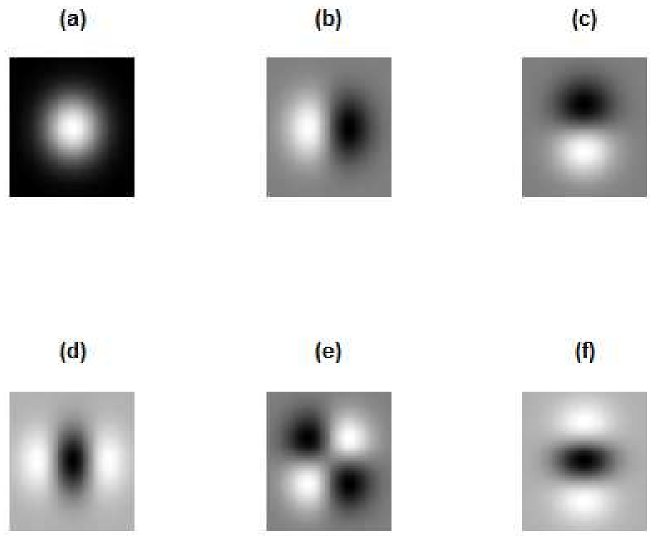


Figure 2.2: Gaussian function and its derivatives illustrations up to second order. Image (a) is G(x,y;t), (b) is ∂_xG(x,y;t),(c) is ∂_yG(x,y;t), (d) is $\partial_{x,x}$G(x,y;t), (e) is $\partial_x\partial_y$G(x,y;t) and (f) is $\partial_{y,y}$G(x,y;t). The brightness (contrast stretching are different)

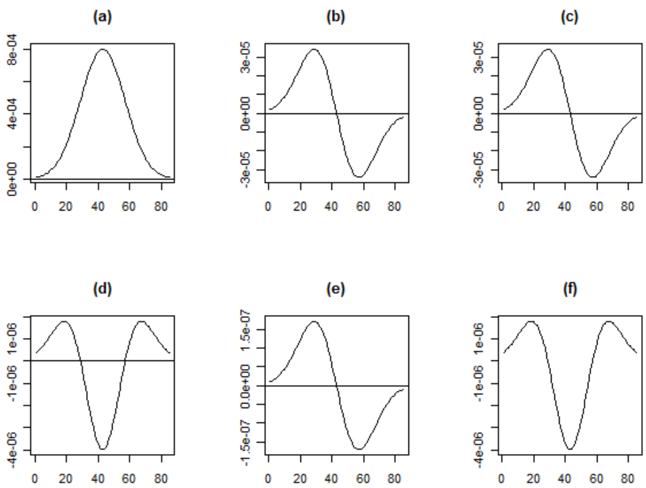


Figure 2.3: Profiles of Gaussian kernel and its derivatives at different orientations. Image (a) is from G(x,y;t),(b) is from ∂_xG(x,y;t),(c) is from ∂_yG(x,y;t), (d) is from $\partial_{x,x}$G(x,y;t), (e) is from $\partial_x\partial_y$G(x,y;t) and (f) is from $\partial_{y,y}$G(x,y;t).

2.6. Image blob

A blob is grey level image region whose background is either bright or dark. Between the grey level region and the background is a gradual transition of the blob into the background. Figure 2.4 shows an example of a bright blob gradually disappearing into the dark background.

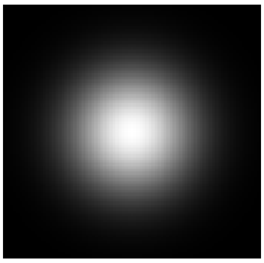


Figure 2.4: Example of an image blob

2.7. Scale-space points

These are points of interest in scale-space, also called critical points. They are points at fixed scale where spatial gradient is zero, ($\nabla L(x, y; t) = 0$), (Kuijper et al., 2001):

$$\frac{\partial}{\partial \mathrm{x}} L(\mathrm{x}, \mathrm{y}; t) = 0; \quad \frac{\partial}{\partial \mathrm{y}} L(\mathrm{x}, \mathrm{y}; t) = 0 \tag{2.7}$$

They include saddle points and extrema (minima or maxima). There two different types of critical points as follows;

Scale-space saddles: A scale-space point is a point whose spatial gradient and scale derivative are zero: $\nabla L = 0$ and $\Delta L = 0$ where ΔL denotes Laplacian (Kuijper et al., 2001).

Top points: They are also called catastrophe points (Kuijper et al., 2001). These are points where the spatial gradient and determinant of Hessian matrix are zero: $\nabla L(\mathbf{x}, \mathbf{y}; \mathrm{t}) = 0$ and $\det \mathbf{H}(\mathrm{x}, \mathrm{y}; t) = 0$. Hessian matrix of a specific scale, t, of second order derivatives in scale-space is defined as follows;

$$\mathbf{H} = \nabla\nabla \mathrm{f} = \begin{pmatrix} \partial_x^2 L(\mathrm{x,y};t) & \partial_\mathrm{x}\partial_\mathrm{y} L(\mathrm{x,y};t) \\ \partial_\mathrm{x}\partial_\mathrm{y} L(\mathrm{x,y};t) & \partial_y^2 L(\mathrm{x,y};t) \end{pmatrix} \tag{2.8}$$

From the scale-space representation, the top points are detected by tracking for critical points where saddle and extrema pair either annihilates or created.

Critical paths: Critical paths are one dimensional curve in scale-space where critical points are traced. They can be found by intersecting the surfaces where gradient is zero from image gradients in both x and y directions. These paths only exist until scale-space points vanish.

2.8. Image minimum variance reconstruction scheme

Kanters et al. (2003b) proposed an image minimum reconstruction scheme based on derivatives of Gaussian filters. Given a set of filters, a general minimal scheme is first derived as follows;

$$S[\bar{f}] \stackrel{\text{def}}{=} \frac{1}{2} \left\|\bar{f}\right\|_{L^2}^2 + \sum_i \lambda_{i\langle f-\bar{f}|\phi_i\rangle} \tag{2.9}$$

Where f is the original image, $\bar{f}$ is the reconstructed image and λ_i are the Lagrange multipliers. According to Kanters, et al, (2003b) the reconstructed image should have the same local derivatives up to the Nth order in the reconstruction points and the variance must be minimal. The first constraint ensures that the reconstruction has the same critical and top points if the orders are N≥1 and N≥2 respectively. The second constraint ensures the reconstructed image is as smooth as possible. This means that the first part satisfies the minimal variance constraint while the second part ensures preservation of features. Taking a functional derivative, the following is obtained:

$$\frac{\delta S[\bar{f}]}{\delta \bar{f}} \stackrel{\text{def}}{=} \bar{f} - \sum_{i=1}^{N} \lambda_i \, \phi_i \qquad (2.10)$$

For unique solution the functional derivative in Equation (2.10) is equated to zero, and it leads to the following

$$\bar{f} = \sum_{i=1}^{N} \lambda_i \, \phi_i \qquad (2.11)$$

Equation (2.11) minimizes the variance if the coefficients λ_i are calculated by substituting it in the following equation

$$\langle f - \bar{f} | \phi_i \rangle = 0 \qquad (2.12)$$

It is evident that the optimal solution depends on the span of filters used to extract linear features of interest. Duits (2005) and Kanters et al. (2003b) developed an image reconstruction algorithm from multi-scale points where an explicit representation up to second order was used for reconstruction. In this research a second order reconstruction case was adopted. The reconstruction algorithm was proposed by Kanters., et al. (2003b) as follows;

$$\bar{f}(x,y) = \sum_{i=1}^{N} a_i \phi_i(x,y) + b_i^{x} \phi_{i,x}(x,y) + b_i^{y} \phi_{i,y}(x,y) + c_i^{xx} \phi_{i,xx}(x,y) + c_i^{xy} \phi_{i,xy}(x,y) + c_i^{yy} \phi_{i,yy}(x,y) \qquad (2.13)$$

This can be shortened as follows for repeated spatial indices,

$$\bar{f}(x,y) = \sum_{i=1}^{N} a_i \phi_i(x,y) + b_i^{\mu} \phi_{i,\mu}(x,y) + c_i^{\mu\rho} \phi_{i,\mu\rho}(x,y) \qquad (2.14)$$

Where; i = 1… N is the enumeration of the points; $\bar{f}$ (x, y) is the reconstructed image; a,b,c are the coefficient vectors of the reconstruction algorithm; $\mu = \{x, y\}$ and $\rho = \{x, y\}$. Equation (2.14) is comparable to first order reconstruction formula, Equation (2.11). As a constraint, all features in every point, i= 1… N, of the reconstructed image should be the same as the original. To ensure that this minimal variance, the following second order constraints are adopted as follows;

$$\langle f - \bar{f} | \phi_i \rangle = 0, \langle f - \bar{f} | \phi_{i,\mu} \rangle = 0 \text{ and } \langle f - \bar{f} | \phi_{i,\mu\rho} \rangle = 0 \qquad (2.15)$$

These constraints can be rewritten as;

$$\langle f | \phi_i \rangle = L_i, \ \langle f | \phi_{i,\mu} \rangle = L_{i,\mu} \text{ and } \langle f | \phi_{i,\mu\rho} \rangle = L_{i,\mu\rho} \qquad (2.16)$$

For all $i = 1, \dots, N$; $\mu = \{x, y\}$ and $\rho = \{x, y\}$. L_i, $L_{i,\mu}$ and $L_{i,\mu\rho}$ are the scale-space representation and its Gaussian blurred derivatives. The reconstruction formula of Equation (2.14), with its constraints and its application is explained further in methodology section. A more in-depth explanation on this reconstruction technique can be found in Kanters., et al. (2003b), Kanters., et al. (2003c), Duits (2005), Kanters, et al. (2005) and Kanters (2007). Based on Equation (2.14), the following linear systems of equation were generated. This was done by substituting Equation (2.14) and (2.16) in Equation (2.15). These linear equations are used for the calculation of the reconstruction algorithm coefficients ***(a, b, c)***.

$$\langle \sum_{i=1}^{N} a_i \phi_i(x,y) + b_i^{\mu} \phi_{i,\mu}(x,y) + c_i^{\mu\rho} \phi_{i,\mu\rho}(x,y) \mid \phi_j \rangle = L_j \tag{2.17}$$

$$\langle \sum_{i=1}^{N} a_i \phi_i(x,y) + b_i^{\mu} \phi_{i,\mu}(x,y) + c_i^{\mu\rho} \phi_{i,\mu\rho}(x,y) || \phi_{j,\gamma} \rangle = L_{j,\gamma} \tag{2.18}$$

$$\langle \sum_{i=1}^{N} a_i \phi_i(x,y) + b_i^{\mu} \phi_{i,\mu}(x,y) + c_i^{\mu\rho} \phi_{i,\mu\rho}(x,y) || \phi_{j,\gamma\omega} \rangle = L_{j,\gamma\omega} \tag{2.19}$$

With$i = 1, \dots, N$; $\mu = \{x, y\}$; $\rho = \{x, y\}$; $\gamma = \{x, y\}$ and $\omega = \{x, y\}$, the repetition use different symbols for same indices have been used to reduce ambiguity e.g. $\phi_{i,\mu\rho}$ where ; $\mu = x, \rho = y$ is $\phi_{i,xy}$. The equations are then simplified to form a system of linear equations as follows;

$$\sum_{i=1}^{N} a_i \langle \phi_i | \phi_j \rangle + b_i^{\mu} \langle \phi_{i,\mu} | \phi_j \rangle + c_i^{\mu\rho} \langle \phi_{i,\mu\rho} | \phi_j \rangle = L_j \tag{2.20}$$

$$\sum_{i=1}^{N} -a_i \langle \phi_{i,\gamma} | \phi j \rangle - b_i^{\mu} \langle \phi_{i,\mu\gamma} | \phi j \rangle - c_i^{\mu\rho} \langle \phi_{i,\mu}\rho_{\gamma} | \phi_j \rangle = L_{j,\gamma} \tag{2.21}$$

$$\sum_{i=1}^{N} a_i \langle \phi_{i,\gamma\omega} | \phi_j \rangle + b_i^{\mu} \langle \phi_{i,\mu\gamma\omega} | \phi_j \rangle + c_i^{\mu\rho} \langle \phi_{i,\mu}\rho_{\gamma\omega} | \phi_j \rangle = L_{j,\gamma\omega} \tag{2.22}$$

The linear algebra can be used to solve equations but a mixed correlation matrix (Kanters et al., 2003b) is required for simplification of the system of equations. The components of the mixed correlation matrix components is a double integration of the respective Gaussian functions e.g. $\langle \phi_i | \phi_j \rangle$ defined as follows

$$\langle \phi_i | \phi_j \rangle = \iint \phi(x\text{-}x_i, y\text{-}y_i, t_i)\, \phi(x\text{-}x_j, y\text{-}y_j, t_j) dxdy \tag{2.23}$$

For each combination of spatial indices, Kanters et al. (2003b) defined the generalized correlation matrix, $\phi\mu_1 \dots \mu_k$, as a $N \times N$ matrix with components $\phi_{ij}, \mu_1 \dots \mu_k = \langle \phi_i, \mu_1 \dots \mu_k | \phi_j \rangle$. The generalized correlation matrix is also known as gram matrix (Duits, 2005). The components of generalized correlation matrix take the following simplified form (Duits, 2005; Kanters et al., 2003b);

$$\phi_{ij}, \mu 1 \dots \mu k = \phi, \mu 1 \dots \mu k \, (\bar{x}, t) |\bar{x} = \bar{x}_{ij}, t = t_{ij} \tag{2.24}$$

With $\overline{x}_{ij} = \overline{x}_i - \overline{x}_j$, $t_{ij} = t_i - t_j$ and for order $0 \leq k \leq 2$. From this definition, Equations (2.20) to (2.22) are rewritten in a the following matrix form

$$\overbrace{\begin{Bmatrix} \phi & \phi_{x} & \phi_{y} & \phi_{xx} & \phi_{xy} & \phi_{yy} \\ -\phi_{x} & -\phi_{xx} & -\phi_{xy} & -\phi_{xxx} & -\phi_{xxy} & -\phi_{xyy} \\ -\phi_{y} & -\phi_{xy} & -\phi_{yy} & -\phi_{xxy} & -\phi_{xyy} & -\phi_{yyy} \\ \phi_{xx} & \phi_{xxx} & \phi_{xxy} & \phi_{xxxx} & \phi_{xxxy} & \phi_{xxyy} \\ \phi_{xy} & \phi_{xxy} & \phi_{xyy} & \phi_{xxxy} & \phi_{xxyy} & \phi_{xyyy} \\ \phi_{yy} & \phi_{xyy} & \phi_{yyy} & \phi_{xxyy} & \phi_{xyyy} & \phi_{yyyy} \end{Bmatrix}}^{\text{Gram matrix}} \times \overbrace{\begin{Bmatrix} a_1 \\ \dots \\ a_{\mathrm{N}} \\ b_1^{\mathrm{x}} \\ \dots \\ b_{\mathrm{N}}^{\mathrm{x}} \\ b_1^{\mathrm{y}} \\ \dots \\ b_{\mathrm{N}}^{\mathrm{y}} \\ c_1^{\mathrm{xx}} \\ \dots \\ c_{\mathrm{N}}^{\mathrm{xx}} \\ c_1^{\mathrm{xy}} \\ \dots \\ c_{\mathrm{N}}^{\mathrm{xy}} \\ c_1^{\mathrm{yy}} \\ \dots \\ c_{\mathrm{N}}^{\mathrm{yy}} \end{Bmatrix}}^{\text{Coefficient vector}} = \overbrace{\begin{Bmatrix} L_1 \\ \dots \\ L_{\mathrm{N}} \\ L_1^{\mathrm{x}} \\ \dots \\ L_{\mathrm{N}}^{\mathrm{x}} \\ L_1^{\mathrm{y}} \\ \dots \\ L_{\mathrm{N}}^{\mathrm{y}} \\ L_1^{\mathrm{xx}} \\ \dots \\ L_{\mathrm{N}}^{\mathrm{xx}} \\ L_1^{\mathrm{xy}} \\ \dots \\ L_{\mathrm{N}}^{\mathrm{xy}} \\ L_1^{\mathrm{yy}} \\ \dots \\ L_{\mathrm{N}}^{\mathrm{yy}} \end{Bmatrix}}^{\text{Feature vector}} \tag{2.25}$$

The solution of Equation (2.25) provides the coefficient vector values for image reconstruction formula, Equation (2.14). Interpretation of this formula is that from every point $\mathrm{i} = 1, \dots, \mathrm{N}$ there are six differentiated Gaussians forming together the reconstructed image. Each point used for reconstruction has six linear systems of equations. The number of linear system of equations to be solved is six times the number of points used. Reconstruction from equidistant and scale-space points were used in this thesis but randomly selected points can be used as well. From Equation (2.25), the feature vector and the mixed correlation matrix is calculated. From this the values of the coefficients vector for each of the point can then be calculated from the linear equations. Feature vector is derived from scale-space representation. With coefficient vectors, reconstruction of an image can be calculated.

2.9. Image resampling

Image resampling is an image processing technique for interpolation of an image into a new image sampling grid. It is a transformation of discrete image information into different samples relative to the original sampling. Image resampling is a technique for manipulating a digital image and transforming it into another form through change of resolution, change of orientation or change of sampling points (Gurjar et al., 2005). Shan et al (2008) described image interpolation as an image operation which estimates a fine resolution image from a coarse resolution image. The process is regarded to as old as computer graphics and image processing (Lehmann et al., 1999). It is required for discrete image manipulations including geometric alignment and registration for image quality improvement. This image processing technique is necessary due to image compression where some pixels or image frames are discarded during encoding process and must be regenerated from the remaining information for decoding or further image analysis (Lehmann et al., 1999). A multitude of existing image interpolation methods includes nearest neighbour, linear, cubic, spline, and the sine function introduced in the 1940's by Shannon (Lehmann et al., 1999). Image re-sampling is a very important image processing technique in computer vision (Lehmann et al., 1999). It is a common image processing technique in medical, industrial and remote sensing applications because during image capture the imaging device imposes quality limitations. Kernel based image resampling involves the use of a kernel function in an image interpolation process.

3. RELATED WORK

This chapter presents related work to this thesis. The first part introduces related multi-scale representation of remotely sensed data. Section 3.2 explains how multi-scale representation can be used for image matching, retrieval of images in large image database suing scale-space approach and image reconstruction. Digital photogrammetric applications have been explained in Section 3.3 while image reconstruction techniques have been discussed in Section 3.4. Relative to a kernel based resampling technique that has been developed, a few related work have been presented in Section 3.5. Finally a summary of the chapter is given at the end of the chapter.

3.1. Related multi-scale representation

Some of existing multi-scale representations include; pyramids, quad tree, wavelets, multi-grids and wedgelets (Lindeberg, 1994). Wavelets are considered a powerful image analysis method used to quantify spatial landscape and plant patterns at multi-scale over vast areas. Among the various uses of wavelets ranges from astronomy to bio-medical imaging, to identify the shape, size and location of individual features of interest. Wavelet techniques are very promising techniques in remote sensing used objectively and automatically quantify ecological features in satellite imagery (Amolins et al., 2007). Quadtree is another multi-scale approach used for image decomposition. An entire image is subdivided depending on the information it holds in various parts (Hossny et al., 2007). Quad tree multi-scale representation takes into account that some regions have more information than others and decomposition depends on the quality criterion. Wedgelets are localized functions at varying scales, locations and orientation that are used for image decomposition at multiple resolutions. They are similar to wavelets only that wedgelets are defined in 2D which allows easy modelling of diagonal lines. Image pyramids is an image representation technique designed to support designed to support efficient image representation (Adelson et al., 1984). It entails a sequence of versions of an original image with resolution reduced at regular steps. This form of degradation is also regarded as a scale-space representation by Stefanidis, et al. (1993).

3.2. Image matching, retrieval and reconstruction

The amount of information contained in scale-space points especially critical points is still an open question in research (Kanters et al., 2003b; Kanters et al., 2003c). Top points are events of topological interest in scale-space representation believed to contain crucial information of image structure (Balmachnova et al., 2005; Duits, 2005; Florack et al., 2000; Kanters et al., 2005). The search of stable features in image representation using scale-space approach has been one of the motivating factors to the use of these points.

Kanters et al. (2003c) developed a content based image retrieval algorithm based on scale-space points. Given an image, Kanters et al. (2003c) algorithm can retrieve the closest matches to the image from a large image database based comparison of scale-space features. Based on the work of Nielsen et al. (2001), Nielsen et al. (2001) investigated what and how much different types of image features can tell about images. Derivatives of Gaussian filters have been found useful in edge detection. Nishihara (1984) developed a stereo matching algorithm using derivative of Gaussian convolutions to generate image regions for matching.

Image reconstruction from differential structure of scale-space points was first proposed by Nielsen et al. (2001). Kanters., et al. (2003b) developed a second order minimal variance reconstruction using multi-scale critical points, presented in more detail in Section 2.8. In Kanters., et al. (2003b) approach, they produced an explicit representation up to the second order using a generalized co-variance matrix. Furthermore this approach was tested by reconstructing an image using random points, equidistant points and scale-space top points. Crowley (2005) also worked on a similar reconstruction approach and argued that a discrete signal can be reconstructed if the basis of function, e.g. Gaussian derivatives, are known. Crowley (2005) argued that a discrete signal can be reconstructed if the basis of function, e.g. Gaussian derivatives, are known. Janssen et al. (2006) describes Kanters et al. (2003b) scale-space image reconstruction as advantageous because of its linearity and the analytical results of the generalized correlation matrix can be found. Duits (2005) and Kanters (Kanters, 2007) also worked on linear minimum variance reconstruction algorithm.

3.3. Scale-space techniques in digital photogrammetry

Scale-space methods are widely applied in digital photogrammetry. In typical applications Stefanidis et al (1993) considers that features represented by the same resolution belong to the same scale-space level. Stefanidis et al (1993), however notes that differential scale variations existing between conjugate features in a number of imagery or various features in a single image is ignored. Scale variations in digital photogrammetric operations have been used between stereo image pairs for image matching. Stereo pair image matching by affine transformation resample one of the two window patches resulting in an image which might belong to the same geometric level of scale-space to the other window patch (Stefanidis et al., 1993).

Scale-space concept is being used in image analysis for object recognition (Forberg, 2007; Heuwold et al., 2007). Heuwold et al. (2007) developed a model for 2D objects extraction by automatic adaptation of landscape object for lower image resolution in a knowledge based image interpretation system. The adaptation process involves scale-space decomposition of the object model from the fine scale into separate parts that can be separately analysed depending on their scale behaviour (Forberg, 2007; Heuwold et al., 2007). This method essentially predicts the objects behaviour in linear scale-space using analysis-by-synthesis and it is developed based on the linking of blob primitives between adjacent scales. Analysis-by-synthesis simulates object appearance in the target scale by generated synthetic images. Predicted object parts in the simulation process are recomposed into a complete object model suitable for extraction in the target resolution. It is noted that prediction is quite straightforward for 1D case but 2D it is sophisticated due to scale-space blob events. Using the same scale-space object prediction principle Heller, et al. (2005) and Vu et al. (2009) developed scale dependent models for feature extraction. An adaptation to the local context of an object in Heller, et al. (2005) model due to ambiguous scale behaviour prediction for image objects is dealt with by adjusting object models for the landscape object of interest and the local context objects separately. Vu et al. (2009) developed a multi-scale approach for building feature extraction based on mathematical morphology from lidar and image data.

Forberg (2007) inspired by scale-space abstraction capability designed an automatic generalization procedure for 3D building models. In 3D models, different level of object representation is useful so as to avoid unnecessary computations. Hao et al. (2008) used scale-space approach to decrease segmentation difficulty in feature extraction using airborne laser scanning data.

3.4. Kernel resampling

There are several image resampling techniques currently available. A kernel based resampling approach which involves the use of a Gaussian function to interpolate new image grid values has been presented in this research. Other mathematical techniques used to create a new version of an image of different sampling than the initial image includes nearest neighbour, bilinear convolution and cubic convolution. Nearest Neighbour is a method where new image grid values are directly derived from the original image by assigning the new grid the pixel values of the nearest pixel. It is considered the simplest resampling method and it does not alter the original values but it has a big risk of losing some pixel values while duplicating others. This method's advantage is its simplicity and the ability to preserve original values in the unaltered scene. Despite this, nearest neighbour's main advantage is the noticeable positional error especially along linear features where there is obvious alignment of features.

Another well known method is bilinear interpolation. This method involves interpolation between the nearest four pixels to the point that best represent that pixel, usually in the middle or upper left of the pixel. The method takes a weighted average of the four pixels in the original image creating entirely new digital values to the new pixel location. From this undesirable results may be achieved if further processing and analysis e.g. classification based on spectral response is to be done. In such a case resampling is normally preferred after classification process.

Cubic interpolation is another image resampling method used to determine the gray levels in an image by a weighted average of 16 closest pixels to the input coordinates. This method is considered to be slightly better than bilinear interpolation and does not possess the disjointed appearance of nearest neighbour resampling. In this method a distance weighted average of a block of sixteen pixels from the original image which surround the new pixel position.

3.5. Summary

To summarize an in-depth understanding of remotely sensed image deep structure is still not a widely explored research area. Data acquisition by remote sensing are stored in form of pixels and this is the simplest unit of analysis widely used by spatial data research. A related approach of analysing images using scale-space features would provide interesting information. Analysis on detection of scale-space point's detection appears unexplored for remote sensing images would provide a different understanding of image representation compared to common fixed scale representations. Tracking of these points in scale-space would provide insights on the structure of satellite imagery. The existing image resampling techniques available are also base on proximity to the new image dimensions. Interpolation of values by kernel based method would provide results based on its symmetry and the flexibility of varying the kernel width. Gaussian symmetry would offer interpolation with more emphasis in the central pixel. Its flexibility of varying Gaussian width appears potential to generate a multi-scale representation similar to scale-space representation. All these have been factored in the forthcoming chapter.

4. DATA AND RESEARCH METHODOLOGY

This chapter starts with description of the dataset then an explanation on the steps adopted in computation scale-space image representation. An approach on scale-space features detection, tracking and analysis is explained in Sections 4.3 and 4.4. An adopted linear image reconstruction algorithm based on scale-space features is described Section 4.5. Development of a kernel based image resampling method and its relation to scale-space representation is presented in Sections 4.6 and 4.7.

4.1. Data sets selection and pre-processing

Both synthetic and remote sensing images were selected for this research. Selection of images was based on different target scene characteristics. Synthetic images believed to provide useful source of information were considered handy in this research. According to Song (2005), synthetic data assist in testing developed algorithm because it offers the option of pre-defining conditions. The use of synthetic images was motivated by the fact that they can be customized to specific geometric features relative to an analysis to be performed with reduced complexities compared to real images. Several synthetic images were simulated in various stages of the thesis for testing the algorithm; nonetheless selections of real images were used in this thesis.

Data selection was based on target scene characteristics and partly due to presence of pure and mixed pixels. Mixed pixels considered in this case as described by Fisher (1997) and shown in Figure 4.1. The different scene characteristics and pixel types consideration was selected for analysis in detection and tracking of scale-space points. According to Tolpekin et al. (2009), boundary pixel presents the easiest case of a mixed pixel for multi-scale analysis. Among the satellite images used are Quickbird panchromatic, Ikonos, Aster and Spot.

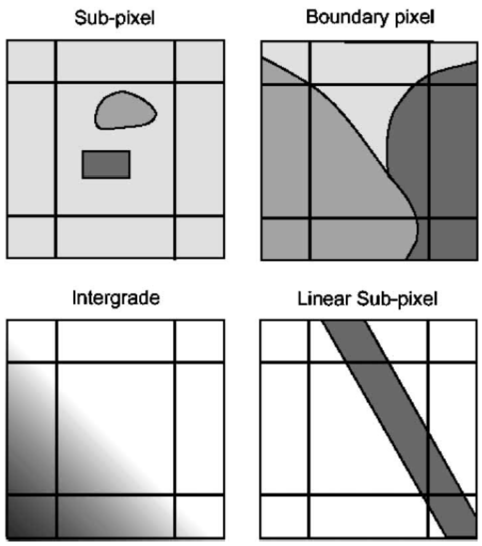


Figure 4.1: Different types of mixed pixels. Source: (Fisher, 1997).

Data preparation was done using ENVI 4.7, ERDAS Imagine 2010 and R a free software environment for statistical computing and graphics environments. R software environment for statistical computing and graphics environments (version 2.11.1) being the main implementation environment. Mathimatica computation software Version 8.0 was also used to understand parts of scale-space implementation as most available tutorial are provided in it (Wolfram Research Inc, 2010). ScaleSpaceViz 1.0 open source software for scale-space visualization software was also used (Kanters et al., 2003a).

4.2. Computation of scale-space representation

A scale-space representation of the selected image using Gaussian function, Equation (2.1) was computed. Computation of images of varying resolution entailed convolution using Gaussian function of varying width. The width of the function was calculated using scale parameter (t) of the Gaussian function, such that increase in scale parameter leads to larger width of the kernel. The relation used for Gaussian kernel width and scale parameter is as shown in Equation (4.1):

$$2 \times 3\sqrt{2t} + 1 \tag{4.1}$$

For every image computation, minimum and maxima scales were selected depending on image size and grey level characteristics. Scale levels were selected from the scale range in two ways for computation of scale-space; linear and logarithmic. Logarithmic scale level selection was used when the target scale-space images were those from finer scales while linear was used when the target was the whole scale range. This was used mainly in the detection of scale-space points for image reconstruction because finer scales the details. Linear scale selection option involved selection of scale levels at equidistant intervals. Six images, L, L_x, L_y, L_{xx}, L_{xy} and L_{yy}, as defined in Section 2.5 are computed at every scale level selected.

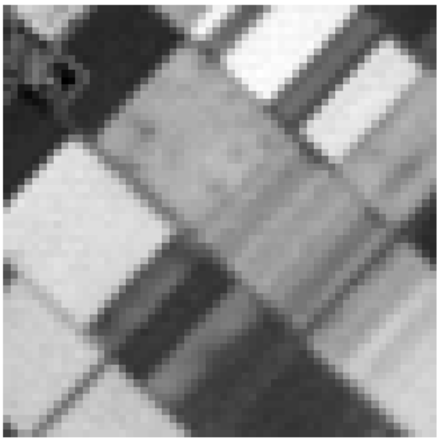


Figure 4.2: An input image for computation of scale-space representation

4.3. Detection and tracking of scale-space points

According to Nielsen, et al. (2001), there are features in an image that contain more image information than others e.g. edges and image blobs. In scale-space representation, scale-space points are believed to contain crucial image information (Kanters et al., 2003b) though it is not known how much information they contain. In practice the points are detected after choosing an appropriate threshold value. This is because the values of the scale-space images have values close to but not equal to zero. Selection of a threshold value was informed by scale-space image histograms, summary information, image characteristics and visual exploration of detected points from various thresholds plotted over scale-space images. Calculation of scale-space location can be achieved from pixel up to sub-pixel accuracy. Sub-pixel accuracy is achieved when zero-crossings is used for calculation. Zero-crossings entail the plotting of the image derivatives as surfaces and interpolation of where the surface crosses the x axis giving the scale space points.

Images of different spatial resolutions were selected based on different image objects and scene characteristics. After setting up a threshold value, scale-space points were calculated for every scale level computed. Calculation of these points means ensuring the conditions given in Section 2.8 hold subject to the threshold set. The threshold value for every scale level was calculated because of the changing image properties due to convolution. Detection of these points is by computation of zero crossings of the first order derivatives, Lx and Ly, calculated in Section 4.2 were used. Scale-space saddles and top points were computed according to their definitions in Section 2.8. Scale-space saddles are computed as points whose gradient either Lx or Ly and the diffusion equation, ∇L, equals to zero. Top points were computed as points whose Lx and Ly are zero and the determinant of its Hessian vanish. The Hessian matrix is calculated using Equation 2.8. For the different types of scale-space points, the same threshold value at every scale level was used.

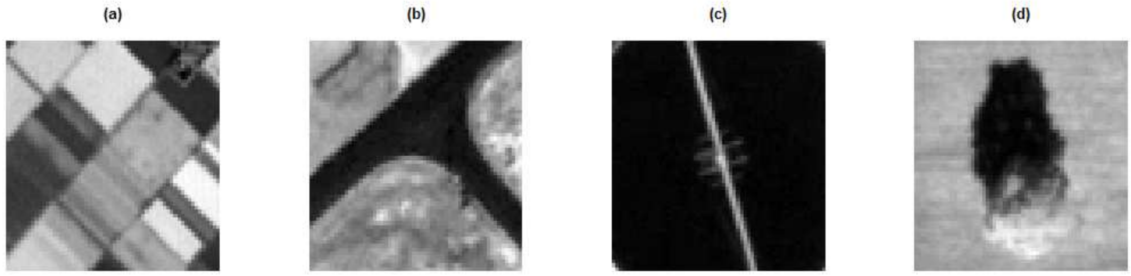


Figure 4.3: Quickbird images used for tracking scale-space top points

Tracking of scale-space points involves computation of these points from scale closer to each other. From the adjacent scales, similar points are identified and a line is drawn to join them. The same process continues in all the computed scale-space representation. The line used to join conjugate points at different scales is the critical path.

4.4. Scale-space points analysis

In detection and tracking of scale-space points, in Section 4.3, some points degenerate 'faster' than others with change of scale. An assumption was made that points which last longer are stronger than those which degenerate at finer scales. Image scenes of different characteristics were selected to analyse the detection and tracking of the points. An analysis on how scale-space points relate to the boundary type of mixed pixel (Fisher, 1997), was performed. This was done in the view that, these points could be having the potential for image reconstruction for scale manipulation. Among the selected scenes included tree with shade and a building roof top containing several chimney and homogeneous backgrounds. From each image detection and tacking of scale-space points was computed. An analysis of detected and tracked points in relation to image objects was then done by visual investigation. Among the selected images for analyses are as shown in Figure 4.3:

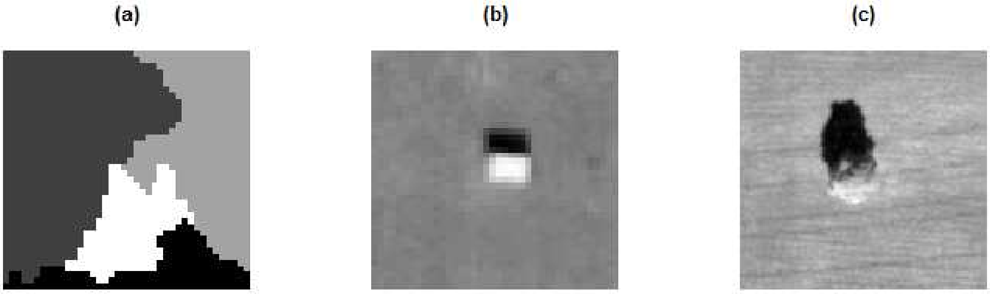


Figure 4.4: Selected images for analysing change of image objects structure in scale-space

Analysis on the assumption regarding the effect of noise in detection and tracking of scale-space points, different strengths of random noise was introduced in Figure 4.3 (a). The analysis of the effect of noise was based on the top point that takes longer to degenerate from each grey level object. The coordinates (x,y,t) of the degeneration point with introduction of different strengths of noise was recorded. The coordinates were then analysed based on scale at which they degenerate and the displacement relative to when no noise is introduced.

4.5. Image reconstruction from scale-space points

The linear systems of Equations (2.20) to (2.22) are represented in the form $\mathrm{Mc} = \mathrm{v}$ as simplified by Equation (2.25). For every selected point for reconstruction, a total of six equations are derived. M is the mixed correlation matrix, c the coefficient vector and v the feature vector. After computation of scale-space representation, the type of reconstruction scale-space points was chosen. Random, equidistant and top points were considered. The feature vector of Equation (2.25) was derived from computed scale-space representation. The values are L, Lx, Ly, Lxx, Lxy and Lyy for every coordinate from all scale levels used in scale-space computation. The values of M are calculated from Equation (2.24) and (2.23). In the computation of the M values, the (x,y,t) coordinates are taken into consideration. The resulting dimension of the M is six times the number of points selected.

Computation of the coefficient vector (c) values is achieved through solving Equation (2.25). Depending on the type of points selected, some values in feature vector are zero. The corresponding rows and columns of the mixed correlation matrix are removed to reduce their effect on other points used. Singular value decomposition was used for calculation of the inverse of M. After the solving for Equation (2.25), the coefficients vector was then substituted in the reconstruction formula Equation (2.13). The same procedure was followed for the three types of points considered. Root mean square error (RMSE) was then calculated between reconstructed and the original image. The RMSE formula is as follows;

$$\mathrm{RMSE} = \sqrt{\frac{1}{NM}\sum_{i=1}^{N}\sum_{i=1}^{M}\epsilon_{ij}^{2}} \tag{4.2}$$

Where $\epsilon_{ij} = f(x_i, y_j) - \bar{f}(x_i, y_j) \quad \forall_i \quad i \in 1 \dots N$ and $\forall_j \quad j \in 1 \dots M$. N and M are the dimensions of the images

4.6. Kernel based Image resampling

A kernel based resampling method based on a Gaussian kernel was developed. The method interpolates the pixel values of a coarse sampling image into a finer sampling. Sampling here is used to refer to the dimensions of an image. Coarse sampling is often also referred to as degradation in this thesis. The input data was achieved by simple averaging of pixels in same neighbourhood to form a discrete level of a pyramid scheme. The Gaussian function used is as shown in Equation (2.1). The degraded image leads to an image of lower sampling relative to the original. In this resampling method the Gaussian function was used to calculate new image grid equal to the original with values at these points calculated depending on the kernel scale parameter. Image difference and ratio of the resampled and original images was used to check the visual quality of the output. They enhance the spectral differences of the compared images. Image difference involved subtraction of values of pixels at conjugate positions. Image ratio entails division of pixel values at equivalent pixel positions in the images used. The resampling process was designed to produce the same dimensions as the original image for this comparison process. Also the RMSE of the pixels excluding the edges was used.

4.7. Scale-space and kernel based resampling parameter relationships

An interpolated image was then compared with the scale-space representation of the same image. In this analysis, the objective was to find a relationship between resampling and scale-space scale parameters. Images from various levels of scale-space representation were compared with resulting resampled images through calculation of RMSE. The comparison involved the exclusion of the pixels along the edges of both images.

The grey level similarity or difference of the images was further tested by visual examination of the image difference and image ratio. The two methods are known to enhance spectral contrasts of images. Image ratio was created by dividing a scale-space image at specific scale by the resulting resampled image. Image ratio is defined as an image enhancement process where DN value of one image is divided by that of another image in a sensor array. Image differencing was done by subtracting the resampled image from the scale-space image. Image ratio results were used to compare the level of agreement of the two images. Image ratios are used in image processing as an effective technique in enhancing or revealing information when there is an inverse relationship between two spectral responses to the same physical phenomena (Mather, 2004). Using the same approach a relationship between the degrading and resampling scales was established. In this case RMSE was calculated against the initial image.

5. RESULTS

This chapter of results is divided into five sections following the order of execution in previous chapter. Section 5.1 presents the computation of scale-space representation while Section 5.2 describes scale-space features. The use of scale-space features for image reconstruction is explained in Section 5.3. A kernel based resampling approach is presented in section 5.4 with its relation to scale-space representation presented in section 5.5.

5.1. Computation of scale-space representation

At every scale level, six differentiated images are computed. Figure 5.1 is a scale-space level computation of an image (Figure 4.2). Image (a) shows a smoothened version of the original image by derivatives of Gaussian function up to second order (Figure 4.2). The sharpness of the image along the edges is lower than the original and the homogeneity within the rectangular patches is higher. Within these smaller patches, the small objects have been smoothened to appear more similar to the background of respective patch. Images (b) and (c) are the gradient images computed by derivatives of Gaussian function in x and y directions respectively. There is an enhancement of edges in x and y directions for images (b) and (c) respectively. Image (d) and (f) form the Laplacian of the image. Image (e) shows enhancement of edges in both directions.

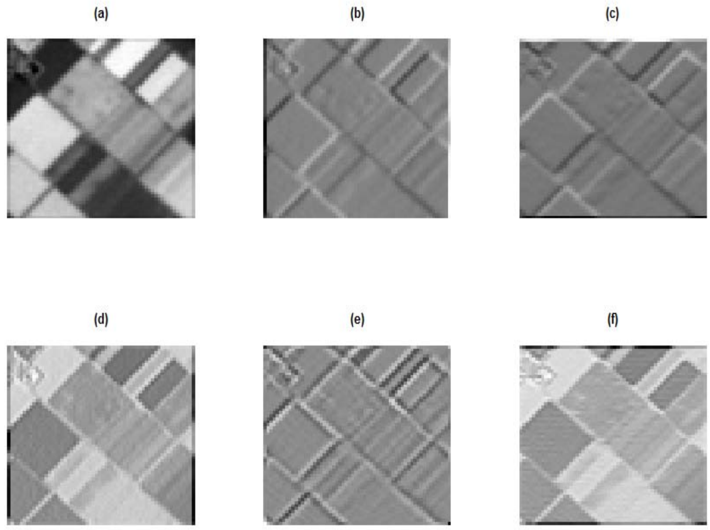


Figure 5.1: Scale-space level images computed for t = 1 for a 60 by 60 pixel image. Image (a) is $L(x,y;t)$, (b) is $L_x(x,y;t)$, (c) is $L_y(x,y;t)$, (d) is $L_{xx}(x,y;t)$, (e) is $L_{xy}(x,y;t)$ and (f) is $L_{yy}(x,y;t)$.

Figure 5.2 shows a scale space representation of Figure 4.2 from several of scale levels. The results depict successive smoothing of features with increase in scale parameter. The convolution of an image results in image blobs of respective objects or its sub-structures present in the image. A blob is grey level homogeneous image region whose background is either bright or dark with a gradually disappearing edge. As the scale parameter increases, a family of blobs around an extremum merge with neighbouring family of blobs to form bigger blobs. From these results the local maxima are manifested on bright blobs while local maxima are on the centroids of dark blobs.

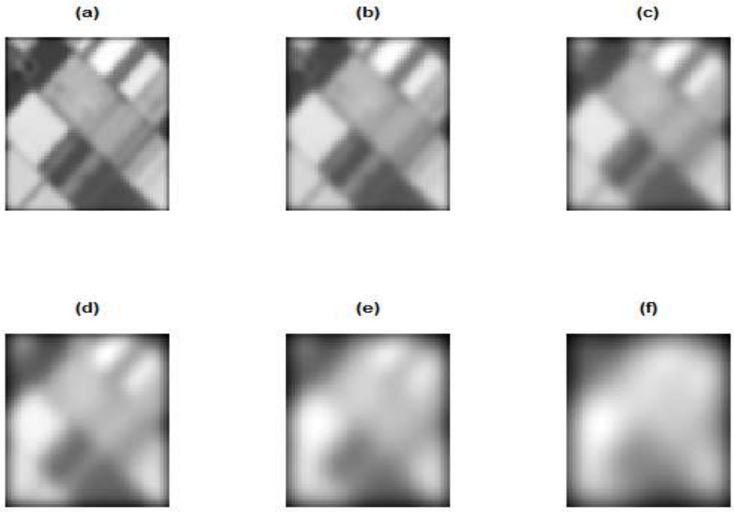


Figure 5.2: Scale-space representation at scale levels t = 0.5,1, 2, 4, 8 and 16. It is a representation of L images only.

5.2. Scale-space points

Images of different characteristics were used for scale-space point detection and tracking and the following findings were observed.

5.2.1. Detection of scale space points

Figure 5.3 is a 216 by 264 Quickbird image plot showing points detected on an image convolved with a scale parameter (t) = 32. The red points are top points, while saddle points are displayed as green and all the plotted points are scale-space points. Figure 5.4 are results of an analysis of scale-space points on different scene characteristics. It is observed from both Figure 5.3 and 5.4 that the points coalesce at the centres of the object blobs. The points also get aligned along the centre of elongated features. Images or image patches with heterogeneous distribution of grey value contribute scattered points in no defined distribution. From large homogeneous image patches points were found to coalesce at the centre of its blob. Object shades were observed to contribute high number of points which is attributed to its homogeneity within the object shade area.

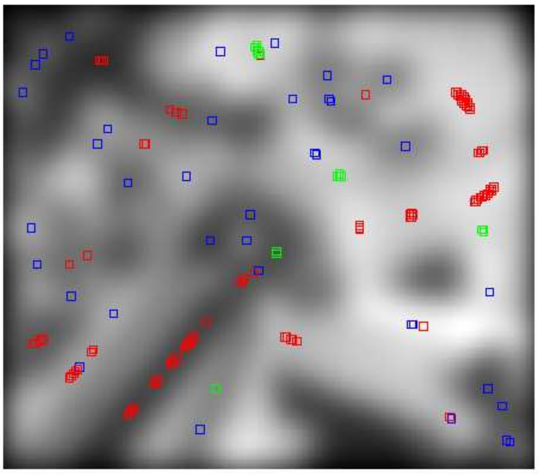


Figure 5.3: Scale-space points computed from a 216 by 264 computed using t = 32 (All points are scale-space points, top points in red and saddles in green)

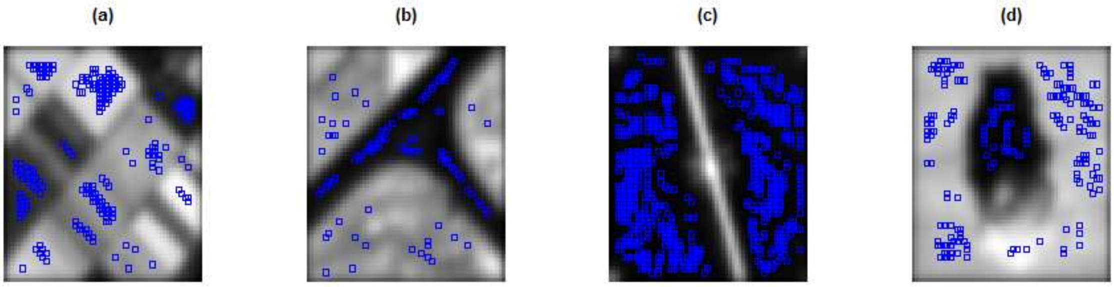


Figure 5.4: Scale-space points detected from t = 1 with threshold values for (a), (b) and (d) is 0.75 and 0.25 for (c).

The number of detected scale-space points reduced with increase in scale parameter as shown in Figure 5.5. The detection of these points with change in scale depends on image characteristics. A large number of points were observed to be detected at finer scales and reduces with increase in scale. Selection of threshold value was observed to be subjective depending on image characteristics especially grey-level variability within the image.

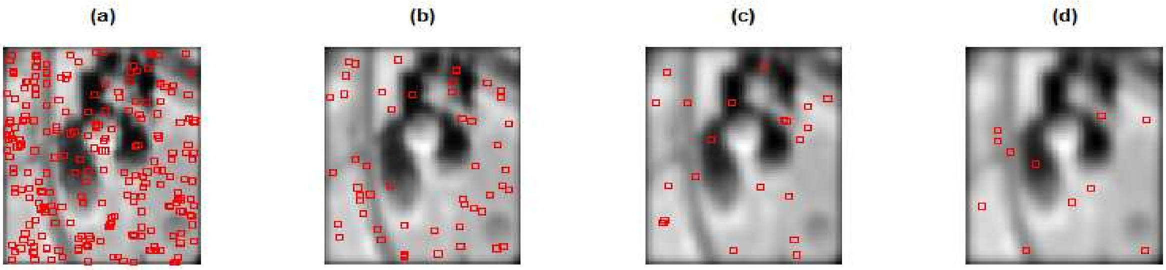


Figure 5.5: Top points from a 100 by 100 image. Computation from scale ranges of $0<t\leq1$, $1<t\leq3$, $3<t\leq6$, $6<t\leq10$ for image (a), (b), (c) and (d) were used respectively. Scale range of 0 to 10 was divided into 100 equidistant steps for calculation of points.

Figure 5.6 is a 3D visualization of top and saddle points from a 200 by 200 pixels image. Top points shown in red while green are scale-space saddle points. 504 points detected for scale parameters between 1 and 20 at 64 scale intervals.

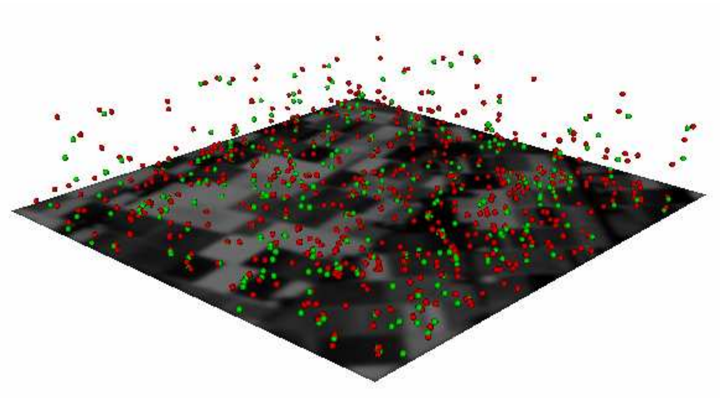


Figure 5.6: 3D visualization of detected scale-space points.

5.2.2. Tracking of scale-space points

Tracking of scale-space points was achieved by linking conjugate points across scale-space representation as explained in Section 4.3. The number of points reduced with coarsening of scale. Figure 5.5 shows the trend of detected points with change of scale. It was aimed at establishing a linkage of points which are topologically equal, for example linking maxima to maxima. Depending on local variations the local minima in the grey level images at each scale were observed to be indicated by dark blobs while local maxima are indicated by bright blobs. It was observed that as scale parameter increases the number of points decreases attributed to suppression of structures.

Figure 5.7 is a 40 by 40 original image and tracking results. It is a Quickbird image of a roof with a chimney. The chimney makes a bright blob with its shade making a dark blob on a heterogeneous background. The second image shows a visualisation of scale-space saddles and top points. Given in red are top points while green are saddle points positioned on the critical paths. The critical curves shown in Figure 5.7 are where the scale-space points are tracked with change in scale parameter. The points detected from the blobs takes longer to degenerate than the points from background.

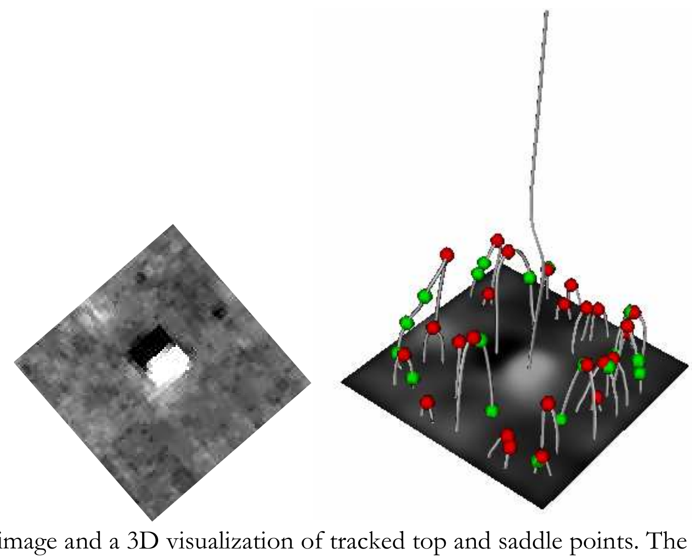


Figure 5.7: An original image and a 3D visualization of tracked top and saddle points. The red points are top points and green points are saddle points.

Scale range	Number of Top points tracked (Np) for Figure 4.3 images			
(t)	(a)	(b)	(c)	(d)
0<t<1	218	140	276	182
1≤t<2	21	23	42	24
2≤t<3	4	12	16	5
3≤t<4	1	5	1	4
4≤t<5	1	4	0	1
5≤t<6	0	0	1	1
6≤t<7	1	2	0	2
7≤t<8	0	2	2	0
8≤t<9	2	2	1	0
9≤t≤10	0	0	2	0
Total	248	190	341	219

Table 1: A tabulation of the number tracked top points with annihilation scale intervals

Table 1 are the results of tracking top points for images in Figure 4.3. All the images are 60 by 60 and a tracking was calculated for 0<t≤10 at step of 0.1. There is a high number of values degenarating at finer scales from 0 to 1. In relation to this, the influence of noise was tested by random noise introduction (Figure 4.4 (a)). Noise on the image was found to influence the presence of points with change in scale. The values given in Table 2 are for points that last longest before they degenerate from the experiments on Figure 4.4 (a) image. The same image was tested to check the influence of noise to all the points. It was observed that in most cases the same points always last longer than other points detected with displacement. The points given in Table 2 are the coordinates of the points which are the last to degenerate than others as influenced by randomly introduced noise (σ). The coordinates at point of degeneration are x,y and scale parameter t. The L features given are the Gaussian convolution and its derivatives up to second order at the vanishing points.

Noise,σ	x	y	t	L	Lx	Ly	Lxx	Lxy	Lyy
30	31.214	26.196	8.436	53.967	0.001	0.003	-0.367	0.015	0.000
25	26.896	27.813	6.944	50.752	0.006	0.003	-0.087	0.060	-0.038
20	28.055	30.926	7.642	40.441	0.002	0.001	-0.077	0.060	-0.046
15	30.907	26.037	6.813	33.926	0.004	0.000	0.000	-0.006	0.187
10	26.193	29.240	6.704	25.477	0.002	0.001	-0.033	0.035	-0.041
5	26.632	31.033	5.824	13.537	0.001	0.000	-0.007	0.013	-0.024
2	29.240	29.929	7.121	7.431	0.000	0.000	-0.008	0.012	-0.015
0	33.570	30.673	4.779	123.920	0.001	0.014	-3.239	-0.185	-0.010

Table 2: Results of tracking a scale-space point with random noise introduced

5.3. Image reconstruction

Using Equation (2.14) image reconstruction can be made from a number of points in scale-space. The points used for reconstruction were selected in three ways. Random, equidistant and scale-space points were the types of points to be used for reconstruction. The feature vector of the selected points was selected from the six differentiated Gaussians and the mixed correlation matrix calculated as given in Equation (2.25). Experimental results from using these three types of points are as follows.

Gaussian blobs were randomly introduced to an image with their centroids used as reconstruction points. Figure 5.8 shows randomly introduced noise and reconstructed image from their centroids. Image (a) is the original image and (b) is the reconstructed image using the centroids of the Gaussian blobs shown in red. Plots (c), (d), (e) and (f) are arbitrarily selected horizontal and vertical profiles selected to assess the level of reconstruction agreement. Image (a) profiles are shown with green line while image (b) profile shown with points. RMSE, using Equation (4.2), of reconstructed image was 0.0012.

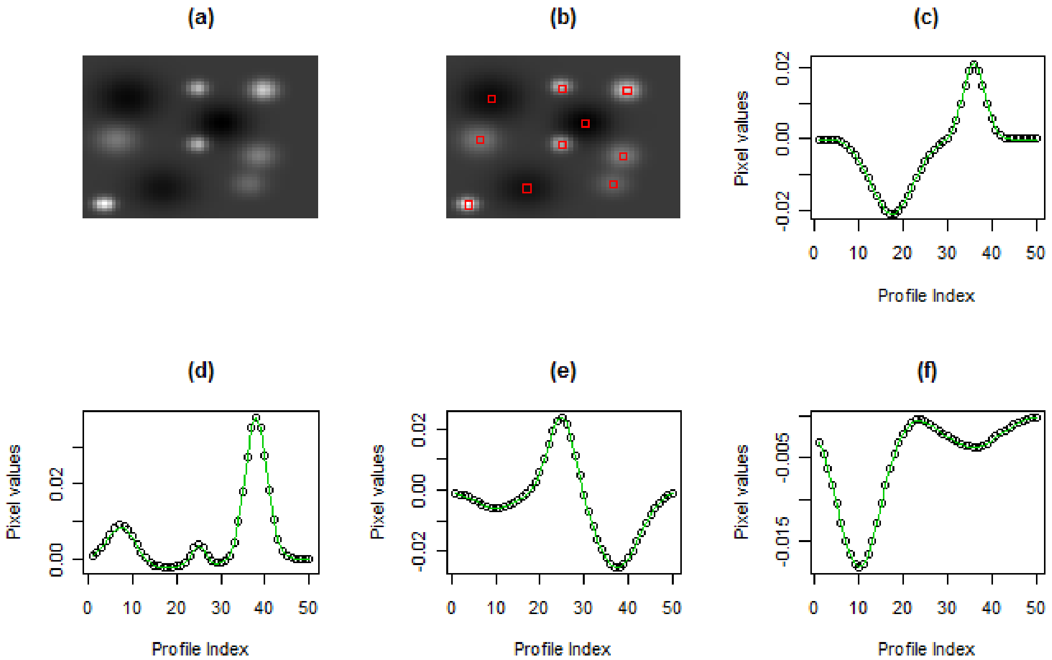


Figure 5.8: Image reconstruction based on Gaussian noise on synthetic image.

From the reconstruction results given in Figure 5.8, the scale-space are used to reconstruct the original image to visually the same as the original. Reconstruction was based on the 10 points centred on the centroids of the Gaussian noise. Differentiated Gaussian images were done over scale parameter range of 1 to 12.5. The level of agreement of the profiles from both the original and the reconstructed is quite good. Equidistant points in scale-space were selected for reconstruction and provided results as shown in Figure 5.9. Reconstruction based scale-space points from fixed scale was also tested. Figure 5.9 shows reconstruction results from 60 by 60 pixel image with Gaussian of standard deviation 20 at the centre shown by image (a). Gaussian derivatives are derived using scale parameter 2. Image plots (b), (c) and (d) have been reconstructed from pixel intervals of 4, 6 and 8 respectively.

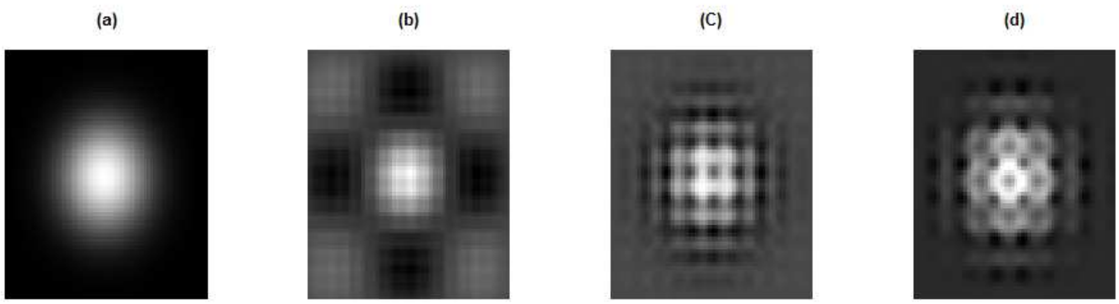


Figure 5.9: Reconstruction of an image with Gaussian at centre.

Table 3 shows the RMSE versus the pixel interval of reconstructed image. The same image in Figure 5.9 (a) has been used for reconstructed images (b), (c) and (d). Figure 5.10 are the plotted results of Table 3. The RMSE graphs in Figure 5.10 shows an asymptotic behaviour as pixel interval decreases.

Pixel steps	No of points	Scale (t) - RMSE		
D	Np	2	4	8
3	399	7857.37200	7931.44800	2108.67700
4	224	1.99647	163182.00000	21625.12000
5	143	0.01639	174.40350	9024.07500
6	99	0.00214	0.78483	40813.01000
7	80	0.00066	0.44771	563.74630
8	63	0.00029	0.00771	10.17586
9	48	0.00018	0.00235	0.63778
10	35	0.00021	0.00096	0.08327
11	35	0.00026	0.00047	0.02067
12	24	0.00031	0.00026	0.00644
15	15	0.00039	0.00024	0.00071
20	8	0.00048	0.00042	0.00027
25	8	0.00045	0.00038	0.00026
28	8	0.00043	0.00035	0.00023
30	3	0.00044	0.00037	0.00026

Table 3: RMSE for image reconstruction from varying pixel steps and scales

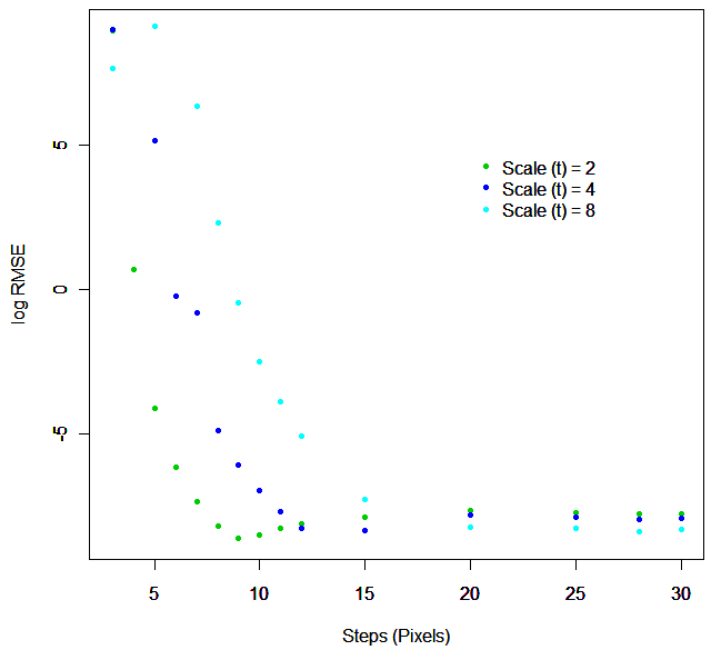


Figure 5.10: RMSE of reconstructed image from equidistant points.

The use of scale-space points produced the same results as those of Figure 5.8. Figure 5.11 (a) shows an image with two Gaussian blobs. Three scale space points are detected in Figure 5.11 (b) at fixed scale of t = 3. The three points reconstructs an image, Figure 5.11 (c), with RMSE of 3.180128.

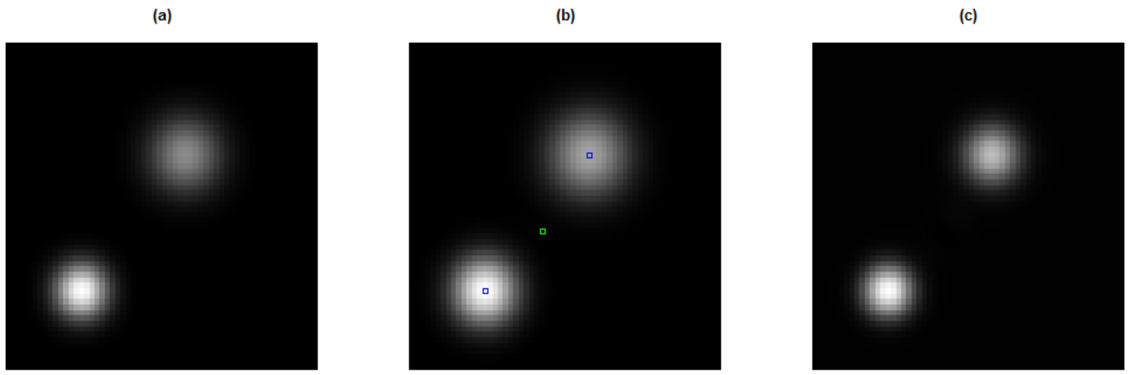


Figure 5.11: Fixed scale image reconstruction using scale-space points.

5.4. Kernel based resampling

The results presented are as explained in Section 4.6. Interpolation to the new image grid depends on the scale parameter of the Gaussian kernel and the resolution of the input image. The scale parameter defines the width of the kernel used and thus defining the coarser grid values for interpolation into the new grid. Degradation scales of 2, 3 and 4 were used. The new image grid was customized to be the same sampling as the original image. Figure 5.12 shows an output, (c), of the interpolation process by scale parameter 1 from a degraded image by scale 2 from an original 60 by 60 pixel image.

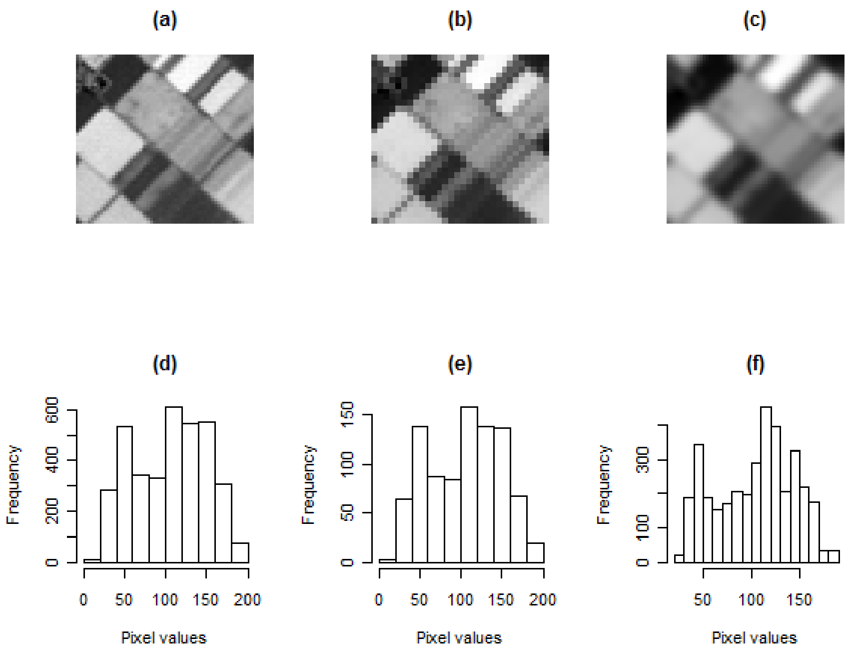


Figure 5.12: Gaussian kernel resampling results. Image (a) is the original 60 by 60 pixel image, (b) is the resampling input image of 30 by 30 pixels and (c) is the resampled results using t=1. Histograms (d), (e) and (f) are for images (a), (b) and (c) respectively.

The similarity in the histograms shows a close approximation of the initial image by the kernel resampling process. Additionally the image difference and ratios between the resampled and original images are as shown in Figure 5.13.

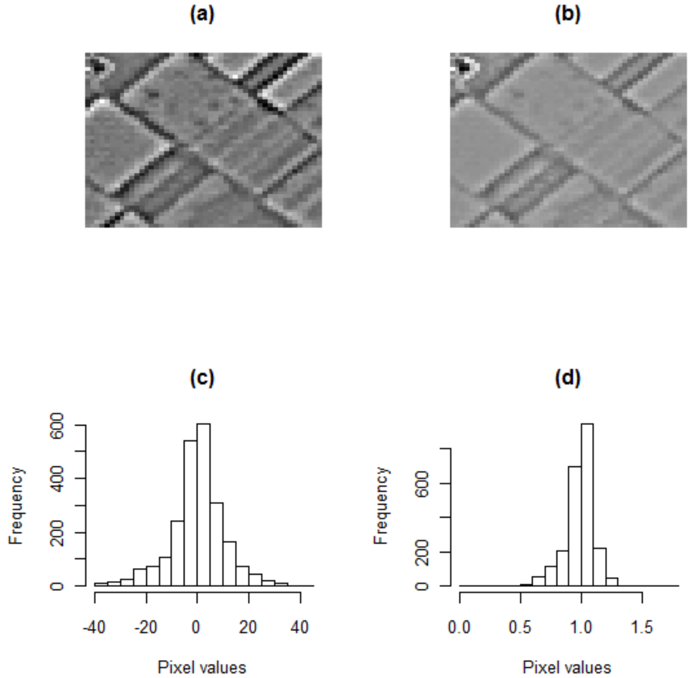


Figure 5.13: Image difference, (a), and ratio, (b), between the original and resampled images. Input of resampled is a degraded image by scale 2 and resample results from t=1. (c) and (d) are histograms of (a) and (b) respectively.

5.5. Scale-space and kernel based resampling parameter relationships

As described in Section 4.7 a relationship between scale-space representation and Gaussian kernel based resampling was established. Table 4 shows results of comparing resampling output from a range of scales against scale-space representation of an image. The resampling experiments were executed for two versions of degraded image by degradation scales of 2 and 3. The values given are RMSE excluding pixels along edges of the resampled image against scale-space level images.

Convolution, t	1	1.25	1.5	1.75	2	2.5	3	1	2	3
Degradation, S	2	2	2	2	2	2	2	3	3	3
Resampling, t	RMSE									
0.25	1.3082	1.6676	1.7341	1.7844	1.8236	1.8804	1.9193	2.5918	2.3838	2.4079
0.5	0.9189	0.637	0.7101	0.7673	0.8125	0.8786	0.9242	1.9980	1.4884	1.5034
0.75	1.2268	0.3004	0.3231	0.3576	0.3913	0.4469	0.4881	1.8874	1.0118	0.9979
1	1.4849	0.4029	0.3495	0.3258	0.3183	0.3248	0.3398	2.0082	0.8935	0.8387
1.25	1.6741	0.5820	0.5117	0.4680	0.4404	0.4115	0.4000	2.1844	0.9788	0.8983
1.5	1.8161	0.7326	0.6586	0.6099	0.5766	0.5361	0.5141	2.3545	1.1247	1.0337
1.75	1.9258	0.8531	0.7782	0.7278	0.6924	0.6475	0.6213	2.5042	1.2727	1.1777
2	2.0127	0.9500	0.8750	0.8240	0.7878	0.7408	0.7125	2.633	1.4076	1.3111
2.25	2.0832	1.0292	0.9543	0.9031	0.8664	0.8184	0.7889	2.7441	1.5270	1.4300
2.5	2.1414	1.0949	1.0203	0.9690	0.9321	0.8834	0.8533	2.8402	1.6322	1.5353
2.75	2.1903	1.1502	1.0758	1.0245	0.9876	0.9385	0.9079	2.9239	1.7250	1.6283
3	2.2318	1.1973	1.1232	1.0720	1.0349	0.9856	0.9547	2.9975	1.8071	1.7109

Table 4: Resampling and scale-space representation comparison

From Table 4 the input images from resampling were generated using degradation scale values. Figure 5.14 are the plotted results in Table 1. From the plot it is evident that resampled images have a high correlation with scale-space images of scale level range from 0.5 to 1.

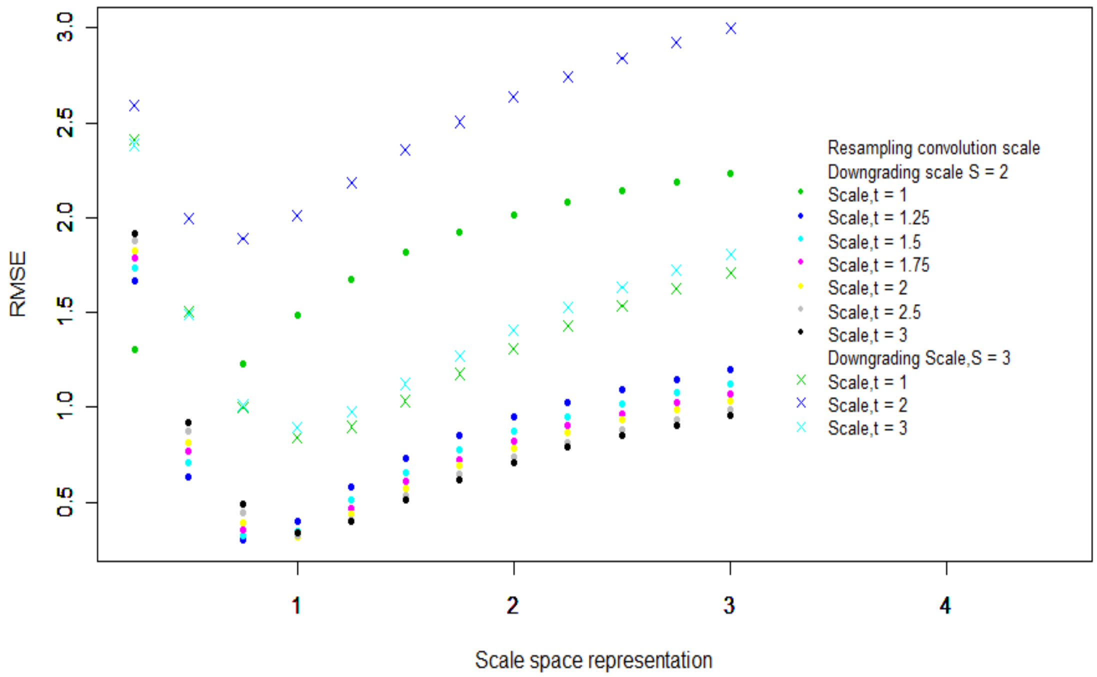


Figure 5.14: Relationship of resample kernel and scale-space scale parameters.

Figure 5.15 are results of a convolved and resampled image with their image difference and ratio. Image (a) represents an image resulting from Gaussian convolution at scale parameter 3 while (b) represents a resampled image to the same dimension as image (a). The input for resampling is a degraded version of image (a) by scale factor 2. Image (c) is the image difference of central pixels of image (a) and (b). The image difference and ratios are a calculation of the areas used for computing the RMSE. Image difference results were used to calculate the RMSE of the cropped part of the image. Image differences with low values of RMSE were observed to be similar to the image ratio plots. Image differencing was used to detect grey level differences of the scale-space image and the resampled image. Its RMSE is 0.832. Image (d) is the image ratio of image (a) and image (b) using same extents as image (c). Figure 5.17 are the RMSE graphs resulting from scale-space image comparison with resampled images. Images (a), (b) and (c) are resampled image results from degradation scale factors of 2, 3 and 4 respectively. Resampled images were compared to images at specific scale-space scale levels as shown by the legends.

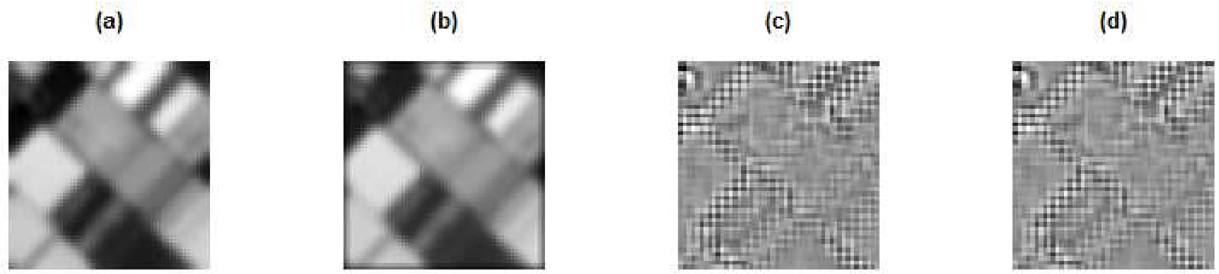


Figure 5.15: Scale-space and resampled image comparison plots.

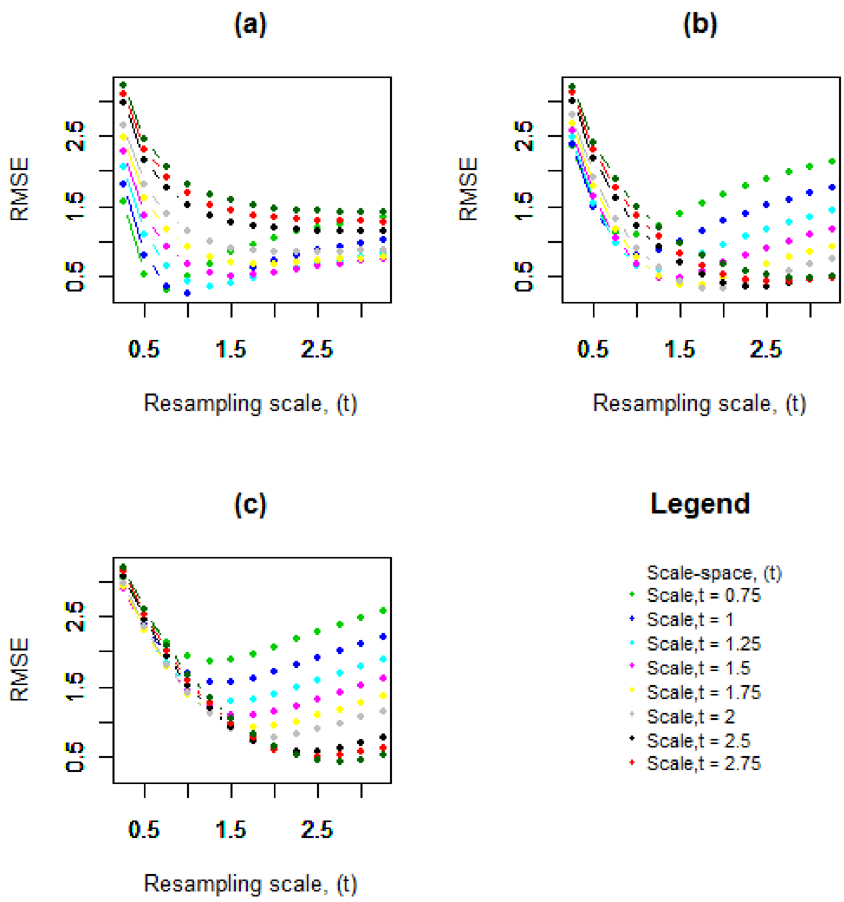


Figure 5.16: Comparison of scale-space representation and results of kernel based image resampling with varying resampling scale parameter (t). Image (a), (b) and (c) are based on resampling input images degraded by S = 2, 3 and 4 respectively

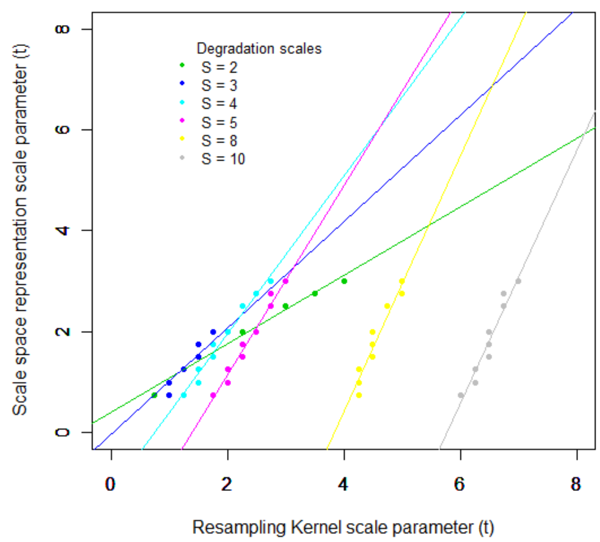


Figure 5.17: Comparison of scale-space and resampling kernel parameters at points of high correlation

From each degradation scale, a relationship between resampling kernel and scale-space scale parameters was established. Every scale-space level was plotted against resampling kernel scale parameter whose output image has the lowest RMSE. For each degradation scale, a linear correlation of the two scale parameters is established. It shows that, a scale-space representation can be approximated using Gaussian kernel based resampling operation. The correlation of resampling results depends on the degradation scale of the input image. As the degradation scale increases the scale parameter of the resampling kernel increases as well forming a parabolic relationship as shown in Figure 5.18.

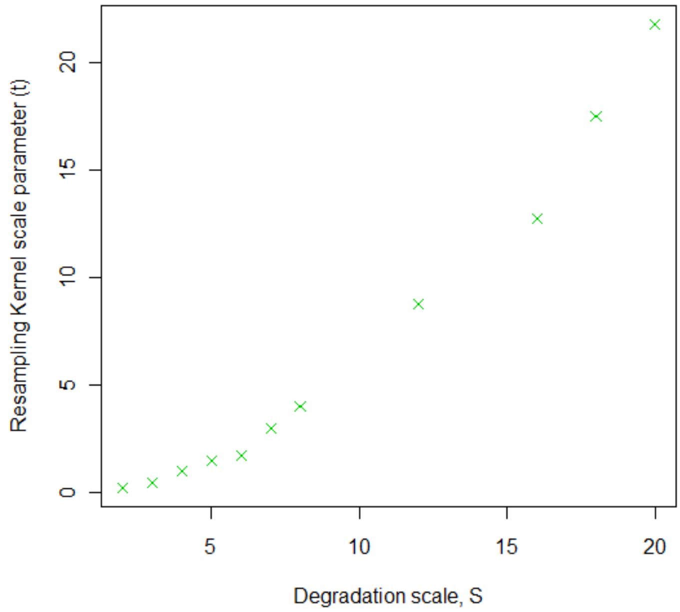


Figure 5.18: Relationship between image degradation scale (S) and resampling scale parameter (t)

6. DISCUSSION

This chapter presents a discussion of methods of implementation and the results of the pre-defined objectives. Image representation is considered a fundamental concern in image processing research because the quality of image interpretation naturally depends on the quality of image representation. Scale-space representation of an image offers a unique approach in multi-resolution image representation. The multi-resolution remotely sensed images have similarity with multi-scale representation generated by scale-space theory. In scale-space representation, the spatial density of image sampling grids is kept constant but the objects resolution in the image changes with coarsening of scale. The constant sampling and symmetry property of the Gaussian kernel makes scale-space representation a unique multi-scale representation. Pyramids representation leads mainly depends on the degradation scale and the initialisation point. Homogeneity is the main factor considered in quadtree representation. Scale-space representation however generates new pixel values taking into account all the directions of the neighbourhood. The flexibility with scale-space representation lies with the ability to choose the width of the convolution depending on the desired resolution.

The implementation of this research was based on both synthetic and single band images. Scale-space analysis based on multi-band satellite images is also feasible. Computation of scale space representation leads to disintegration of image structures manifested as independent entities in form of blobs. Blobs monotonically merges with neighbouring family of blobs to form bigger blobs (Figure 5.1) with the coarsening of scale. In the computed representation features in form of blobs get suppressed at varying rates across the computed scale range. For analysis on the suppression of image structures, scale-space points were detected from the computed scale-space representation. They are extrema and saddle points detected from first and second order image derivatives. Gaussian functions' first derivatives convolve the images to produce scale-space images whose peaks are edges while for the second derivatives zero-crossings correspond to the edges of the image. As the scale coarsens, the image pixel values tend towards zero. This trend influences the selection of threshold value for detection of points for an image at as the scale coarsens. Threshold value selection for every image is varies depending on image characteristics and its derivatives statistical information. It was observed from Figure 5.4 that images with mostly homogeneous image patches require finer threshold value than heterogeneous images. From the experiments most images the maximum threshold value used was 0.75. It is however noted that threshold value selection is a subjective process mainly depending on the image characteristics and analyst requirements.

A further analysis of on location of scale-space points based on scene characteristics and tracking of top points was performed. From Figure 5.4, it was scale-space points coalesce at the centroids of image blobs. A dark blob contributes more points than the brighter blobs. This can be attributed to homogeneity of the blobs most of which were as a result of shading effects. On heterogeneous image or image patches, scale-space points are distributed in no defined pattern. The geometry of features in an image influences the alignment of the points. Figure 5.3, 5.4 and 5.5 confirms this trend and this can be attributed to high intensity gradients along the edges where few scale-space points are detected. The trend of reduction in heterogeneity of an image with coarsening of scale consequently led to successive decrease in detected points. An analysis on the influence of noise with coarsening of scale could provide an alternative of image analysis where noise influence is reduced as compared to when single image is used. To analyse scale-space points with change of scale, same size images (Figure 4.3) of different scenes gave a similar trend of point annihilation. From the results in Table 4, over 70% of the scale-space points annihilate at fine scales $0 < t \leq 1$. This characteristic is attributed to image grey level variability at finer scale than coarser scales. This observation is related Marr et al. (1980) argument that raw grey-levels contain a lot of redundant information about an image surface brightness. The tracking of scale space points confirms Lindeberg's (1994) idea that a non-degenerate scale-space point present at a scale level can be traced to a similar scale-space point at slightly coarser and slightly finer scale. Complication is faced when the blobs merge and where there is creation of new points. This concurs with Kuijper, et al (2002) observation that the creation of new points disrupts a good linking of extrema across different scales. The behaviour of a scale-space blob is solely determined by the critical path during a short scale interval around the bifurcation moment (Lindeberg, 1994). Noise influence on detection and tracking of points showed a destabilizing effect with scale-space points lasting longer with introduction of noise, Table 2. Random noise displaces degeneration point in no defined direction and with stronger noise there is a tendency of a point to degenerate at coarser scales.

As part of analysis scale-space features of satellite imagery, a minimum variance reconstruction algorithm proposed by Kanters et al. (2003b) was adopted. Three different types of scale-space points were used; random, equidistant and top points of different features. Based on random points used in Figure 5.8, the algorithm reconstructed a synthetic image to RMSE value of 0.0012. The profiles for the original and reconstructed images show a close approximation by the algorithm. Reconstruction based on equidistant points depends on pixel interval of the selected points. Figure 5.9 shows the results when scale-space features from a fixed scale are used for reconstruction. Singular value decomposition was used calculation of the inverse of the gram matrix because of the very small values it contains. Reconstruction for remotely sensed images produced weak results and it is recommended for further research. It was also noted that addition of features from higher order image derivatives may improve the reconstruction ability of the algorithm because it only depends on differentiated Gaussians. Other image reconstruction algorithm based on scale-space features of an image may be existing.

The kernel based resampling produces a blurred approximation of an original image from its degraded version (Figure 5.12). The image interpolation can be considered qualitative compared to nearest neighbour, cubic and bilinear interpolation due to its symmetry, emphasize that is gives to neighbouring pixels than distant ones and the ability to vary the kernel width. It is also noted that the type of image representation of the input image influences the results, averaging of neighbouring pixels was applied for input images and a different way of computing input data could provide varying results though a similarity is expected. And as given in Figure 5.17, linear correlations exist between resampled images and scale-space representation. A Gaussian kernel based interpolation can be used in the approximation of scale-space representation of an image.

7. CONCLUSION AND RECOMMENDATIONS

The main objective of this research was to explore potential of scale-space theory for satellite image analysis. Both synthetic and real satellite images were used and the research questions were answered with the following conclusion and recommendations.

7.1. Conclusions

Remotely sensed image analysis based on scale-space theory provides a unique multi-scale image representation for image processing. It offers a hierarchical representation of image and a derivation of scale-space features from its derivatives. Existing scale-space representation involves convolution of images with image dimension maintained. From this thesis a scale-space representation can also be built by Gaussian kernel interpolation of original image values into a finer image grid by varying the kernel width. Gaussian based resampling offers a flexible approach for image interpolation due to the freedom of setting the kernel width relative to input data and user requirements.

Scale-space points detection is influenced by image objects and the edges. The variation of grey level values of image objects determines the spatial distribution of the points. The points are located at the centre of image blobs and edges. The points get aligned with respect to objects geometry. Scale-space saddles are mostly detected along the edges of an image. The effect of illumination leads to object shades which act as homogenous objects in an image. Tracking of points in scale-space follows critical path. With coarsening of scale some points are created while existing ones degenerates. Degeneration is attributed to suppression of image structures with change of scale. The threshold value for scale-space point detection decreases with coarsening of scale. Structures which last longer in scale-space representation have scale-space points lasting longer as well. Presence or random introduction of noise in an image destabilises image structures in no defined direction. The points last longer with stronger noise than when there is weak or no noise.

Reconstruction based on scale-space points is not sufficient for high quality reconstruction. Minimum variance reconstruction algorithm works well with simple images but not very high resolution images. The loss of important information at 'very' fine scales could be one of the problems. Introduction of features from higher order derivatives may be one possibility to improve the quality of results. Analysing images in a multi-resolution format provides an easier to exploit version of an image than single resolution. Map making from remote sensing images could benefit from hierarchical decomposition of images for different map scale out puts.

7.2. Recommendation

The following recommendations are made for further research;

- Exploration of scale-space theory analysis based on spectral and spatial properties of the remote sensing imagery
- Object oriented analysis depends on segmentation. The first order scale-space image derivatives can potentially provide a different aspect of image segmentation in object oriented analysis
- Investigation on the possible use of scale-space points for photogrammetric image matching.

- Further research on the minimum variance reconstruction algorithm based on satellite images and including features higher order image derivatives.

APPENDIX: SOURCE CODE

```
#This is the source code for this thesis implementation
#This only includes the main parts of implementation prepared in R environment
#################################################################
#Initialisation part of the code
 #The information provided in this section are initialisation part of code to assist in code understanding
Path <- 'D:\\Working folder\\' #
dir.create(Path, recursive = TRUE)
B <- 'filename.txt'          #file name
temp <- read.table(paste(Path,B,sep=""),skip=5)   #load image array
d<-dim(temp) #dimensions of the image array
M <- d[1]
N <- d[2]
x <- 1:M
y <- 1:N
A <- array(0,c(M,N))
A[,]<-data.matrix(temp)
#Transforming image array in R (rotation)
z <- matrix(as.vector(A), ncol=N,nrow=M, byrow=TRUE)
A[,] <- t (z[M:1,N:1])[M:1,]
#Normalization of image values
A <- A - min(A)
A <- 255*A/max(A)
#############################################################
#Setting of parameters
#For reference from various parts of the source code
#Random values (Given to assist in interpretation of source code)
Np <- 10 #Number of points
xb <- c(10,15,76,60,72,78,50,50,35,20)
yb <- c(10,50,40,60,23,80,81,47,20,75)
tb <-  c(log((2:(Np+1))))
Ib <- c(1.5,2,1.5,-3,1,2,1,1,-3,-4)
Ns <- Np
Scales <- array(0,Ns)
Scales <- tb
#######################################################
#Image degradation (Through pixel averaging)
#######################################################
#Degraded image parameters
S<-2 #degradation scale
Mc <- M/S
Nc <- N/S
xc <- 1:Mc
yc <- 1:Nc
Ac <- array(0,c(Mc,Nc)) #Degraded image dimension
for(i in 1:Mc)
```

```
for(j in 1:Nc)
(
  Ac[i,j]<- mean(A[((i-1)*S+1):(i*S),((j-1)*S+1):(j*S)])
)
############################################
#Kernel Based Resampling (Gaussian)
############################################
Ar <- array(0,c(M,N))
Gf <- function(t,x,y)
(
 val <- x^2+y^2

 if(t==0)
 (
        val[val!=0] <- -1
        val <- val +1
 ) else  val <- exp(-val/(4*t)) / (4*pi*t)

 return(val)
)
t <- 0.5*S^2
for(i in 1:M)
for(j in 1:N)
(
 xi <- i-1/2
 yi <- j-1/2
 ic <- ceiling(xi/S)
 jc <- ceiling(yi/S)
 i1 <- max(c(ic-1,1))
 i2 <- min(c(ic+1,M/S))
 j1 <- max(c(jc-1,1))
 j2 <- min(c(jc+1,N/S))
 xc <- ((i1:i2)-0.5)*S
 yc <- ((j1:j2)-0.5)*S
 N1 <- length(xc)
 N2 <- length(yc)
 darr <- Ac[i1:i2,j1:j2]
 xdiff <- xc-xi
 ydiff <- yc-yi
 warr <- array(0,c(N1,N2))
 for(k in 1:N1)
 for(l in 1:N2)
  warr[k,l] <- Gf(t,xdiff[k],ydiff[l])
 warr <- warr/sum(warr)
 Ar[i,j]<- sum(warr*darr)
)
```

```
#Rmse of the image resampled Vs convolved image
rmse <- (sum((Ar[(M/10+1):(M-(M/10)),(N/10+1):(N-(N/10))]-Lt[(M/10+1):(M-(M/10)),(N/10+1):(N-(N/10))])^2)/(((M-(2*M/10))*(N-(2*N/10)))))^0.5

x11()
par(mfrow=c(2,2))
#Image difference calculation; Edge pixels excluded
Adiff<- Lt[(M/10+1):(M-(M/10)),(N/10+1):(N-(N/10))]-Ar[(M/10+1):(M-(M/10)),(N/10+1):(N-(N/10))]
#Image ratio calculation; Edge pixels excluded
Aratio<- Lt[(M/10+1):(M-(M/10)),(N/10+1):(N-(N/10))]-Ar[(M/10+1):(M-(M/10)),(N/10+1):(N-(N/10))]
image(x,y, Ar, col=gray((0:255)/255), main = paste('Resampled, Rmse = ',round(rmse,3)), xlab='',ylab='',axes=FALSE)
image(1:(M-(2*M/10)),1:(N-(2*N/10)), Adiff, col=gray((0:255)/255), main = paste('Difference, tr = ', tr,sep=''), xlab='',ylab='',axes=FALSE)
image((M/10+1):(M-(M/10)),(N/10+1):(N-(M/10)), Aratio, col=gray((0:255)/255), main = paste('Image Ratio, tc = ', t,sep=''), xlab='',ylab='',axes=FALSE)

#################################################################
# Components of convolution kernel and its derivatives
#################################################################
#Gaussian function
G <- function(t,u,v)
(
 val <- 1/(4*pi*t)*exp(-(u^2+v^2)/(4*t))
 if(t==0) val <- 0
 return(val)
)
#Gaussian function first derivative w.r.t. x
Gx <- function(t,u,v)
(
 val <- 0
 if(t>0) val <- -u*G(t,u,v)/(2*t)
 return(val)
)
#Gaussian function first derivative w.r.t. y
Gy <- function(t,u,v)
(
 val <- 0
 if(t>0) val <- -v*G(t,u,v)/(2*t)
 return(val)
)
#Gaussian function 2nd derivative w.r.t. x
Gxx <- function(t,u,v)
(
 val <- 0
```

```
  if(t>0) val <- (((u/(2*t))^2)-1/(2*t))*G(t,u,v)
  return(val)
)
#Gaussian function 2nd derivative w.r.t. x and y
Gxy <- function(t,u,v)
(
  val <- 0
  if(t>0) val <- ((u*v)/((2*t)^2))*G(t,u,v)
  return(val)
)
#Gaussian function 2nd derivative w.r.t. y
Gyy <- function(t,u,v)
(
  val <- 0
  if(t>0) val <- (((v/(2*t))^2) - 1/(2*t))*G(t,u,v)
  return(val)
)
#Array of 0 values with dimension M*N for coarse image
#Note: same spatial sampling as fine resolution image
L   <- array(0,c(M,N,Ns))
Lx  <- array(0,c(M,N,Ns))
Ly  <- array(0,c(M,N,Ns))
Lxx  <- array(0,c(M,N,Ns))
Lxy  <- array(0,c(M,N,Ns))
Lyy  <- array(0,c(M,N,Ns))
#################################################################
##Building scale space representation (Over a range of scales)
#################################################################
for(ls in 1:Ns)
(
   s <- Scales[ls]
          #window size with 2s as the variance of gaussian kernel
   W <- round(3*sqrt(2*s))
   WS <- 2*W+1
   Gs <- array(0,c(WS,WS)) # Array of 0 values with dimension WS*WS
   Gsx <- array(0,c(WS,WS)) # Array of 0 values with dimension WS*WS
   Gsy <- array(0,c(WS,WS)) # Array of 0 values with dimension WS*WS
   Gsxx <- array(0,c(WS,WS)) # Array of 0 values with dimension WS*WS
   Gsxy <- array(0,c(WS,WS)) # Array of 0 values with dimension WS*WS
   Gsyy <- array(0,c(WS,WS)) # Array of 0 values with dimension WS*WS
   for(i in 1:WS)
   for(j in 1:WS)
   (
     Gs[i,j]    <-   G(s,i-W-1,j-W-1)
     Gsx[i,j]   <-  Gx(s,i-W-1,j-W-1)
     Gsy[i,j]   <-  Gy(s,i-W-1,j-W-1)
     Gsxx[i,j]  <- Gxx(s,i-W-1,j-W-1)
```

```
    Gsxy[i,j]  <- Gxy(s,i-W-1,j-W-1)
    Gsyy[i,j]  <- Gyy(s,i-W-1,j-W-1)
  )
  Gs[,] <- Gs[,]/sum(Gs[,])
  # Add zero rows and columns
  #Offsetting the image array by 2*w zero values
  Atemp <- array(0,c(M+2*W,N+2*W))
  #Assigning values of A into the central pixels Atemp (Makes convolution faster)
  Atemp[(W+1):(W+M),(W+1):(W+N)] <- A[,]
  for(i in (1+W):(W+M)) #Atemp Columns
  for(j in (1+W):(W+N)) #Atemp Rows
  (
    L[i-W,j-W,ls]   <- sum(Atemp[(i-W):(i+W),(j-W):(j+W)] * Gs)
    Lx[i-W,j-W,ls]  <- sum(Atemp[(i-W):(i+W),(j-W):(j+W)] * Gsx)
    Ly[i-W,j-W,ls]  <- sum(Atemp[(i-W):(i+W),(j-W):(j+W)] * Gsy)
    Lxx[i-W,j-W,ls] <- sum(Atemp[(i-W):(i+W),(j-W):(j+W)] * Gsxx)
    Lxy[i-W,j-W,ls] <- sum(Atemp[(i-W):(i+W),(j-W):(j+W)] * Gsxy)
    Lyy[i-W,j-W,ls] <- sum(Atemp[(i-W):(i+W),(j-W):(j+W)] * Gsyy)
  )
#Plotting of scale space level images
  x11()
  par(mfrow=c(2,3))
  image(x,y, L[,,ls], col=gray((0:255)/255), main=paste('L, t=',s,sep=''),xlab='',ylab='')
  image(x,y, Lx[,,ls], col=gray((0:255)/255), main='Ltx',xlab='',ylab='')
  image(x,y, Ly[,,ls], col=gray((0:255)/255), main='Lty',xlab='',ylab='')
  image(x,y, Lxx[,,ls], col=gray((0:255)/255), main='Ltxx',xlab='',ylab='')
  image(x,y, Lxy[,,ls], col=gray((0:255)/255), main='Ltxy',xlab='',ylab='')
  image(x,y, Lyy[,,ls], col=gray((0:255)/255), main='Ltyy',xlab='',ylab='')
)
##################################################################
#Calculation of Scale space points
#Matrix of 2nd order derivatives in scale space (Hessian Matrix)
H <- array(0,c(M,N,2,2))  #Hessian Matrix
H[,,1,1] <- Ltxx
H[,,1,2] <- Ltxy
H[,,2,1] <- Ltxy
H[,,2,2] <- Ltyy
eps <- 1.5e-1     #Threshold for image values

DH <-array(0,c(M,N))
for(i in 1:M)
for(j in 1:N)
(
  DH[i,j] <- det(H[i,j,,])
)

#Catastrophe points (Top points)
```

```
Pts <-which((abs(A)>eps)&(abs(Ltx)<eps)&(abs(Lty)<eps)&(abs(DH)<eps),arr.ind=TRUE)
#Scale space saddle points
Ssp <-which((abs(A)>eps)&(abs(Ltx)<eps)&(abs(Lty)<eps)&(abs(Ltxy)<eps)&(abs(Ltyy)<eps),
arr.ind=TRUE)
#Spatial critical points
Scp <- which((abs(A)>eps)&(abs(Ltx)<eps)&(abs(Lty)<eps),arr.ind=TRUE)
#Plotting of scale-space points
x11()
par(mfrow=c(2,2))
image(x,y, Lt, col=gray((0:255)/255), main=paste('Critical points, W=',W,sep=''),xlab='',ylab='')
points(Scp,col="blue", pch=22)
image(x,y, Lt, col=gray((0:255)/255), main=paste('Catastrophe points, W=',W,sep=''),xlab='',ylab='')
points(Pts,col="red", pch=22)
image(x,y, Lt, col=gray((0:255)/255), main=paste('Saddle points, W=',W,sep=''),xlab='',ylab='')
points(Ssp,col="green", pch=22)
image(x,y, Lt, col=gray((0:255)/255), main=paste('Scale space points'),xlab='',ylab='',axes=FALSE)
points(Scp,col="blue", pch=22)
points(Pts,col="red", pch=22)
points(Ssp,col="green", pch=22)

#################################################################
# Components of mixed correlation matrix (gram matrix)
#################################################################
FI <- function(t,x,y)
(
 #val <- 1/(4*pi*t)
 val <- exp(-(x^2+y^2)/(4*t)) / (4*pi*t)
 if(t==0) val <- 0
 return(val)
)
FIx <- function(t,x,y)
(
 val <- 0
 if(t>0) val <- -x*FI(t,x,y)/(2*t)
 return(val)
)
FIy <- function(t,x,y)
(
 val <- 0
 if(t>0) val <- -y*FI(t,x,y)/(2*t)
 return(val)
)
FIxx <- function(t,x,y)
(
 val <- 0
 if(t>0) val <- (((x/(2*t))^2)-(1/(2*t)))*FI(t,x,y)
 return(val)
```

```
)
FIxy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- ((x*y)/((2*t)^2))*FI(t,x,y)
  return(val)
)
FIyy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (((y/(2*t))^2)-(1/(2*t)))*FI(t,x,y)
  return(val)
)
FIxxx <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (-(x^3)/(8*(t^3))+3*x/(4*(t^2)))*FI(t,x,y)
  return(val)
)
FIxxy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (-((x^2)*y)/(8*(t^3))+y/(4*(t^2)))*FI(t,x,y)
  return(val)
)
FIxyy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (x/(4*(t^2))-x*(y^2)/(8*(t^3)))*FI(t,x,y)
  return(val)
)
FIyyy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (-(y^3)/(8*(t^3))+3*y/(4*(t^2)))*FI(t,x,y)
  return(val)
)
FIxxxx <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- ((x^4)/(16*(t^4))-3*(x^2)/(4*(t^3))+3/(4*(t^2)))*FI(t,x,y)
  return(val)
)
FIxxxy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (((x^3)*y)/(16*(t^4))-3*x*y/(8*(t^3)))*FI(t,x,y)
```

```
  return(val)
)
FIxxyy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- (((x^2)*(y^2))/(16*(t^4))-x^2/(8*(t^3))-y^2/(8*(t^3))+1/(4*(t^2)))*FI(t,x,y)
  return(val)
)
FIxyyy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- ((x*(y^3))/(16*(t^4))-3*x*y/(8*(t^3)))*FI(t,x,y)
  return(val)
)
FIyyyy <- function(t,x,y)
(
  val <- 0
  if(t>0) val <- ((y^4)/(16*(t^4))-3*(y^2)/(4*t^3)+3/(4*(t^2)))*FI(t,x,y)
  return(val)
)
#################################################################
##Calculation of mixed correlation matrix values (MCM)
#################################################################
#Parameters of the MCM
Arr <- array(0,c(Np,2))
xp <- array(0,Np)# x coordinates
yp <- array(0,Np)# y coordinates
tp <- array(0,Np)# list of scale parameters
#Paramaters being replicated
xp <- xb+0*rep(1,Np)
yp <- yb-0*rep(1,Np)
tp <- tb
Arr[,1] <- xp
Arr[,2] <- yp

#################################################################
MCM <- array(0,c(6*Np,6*Np))#Initializing MCM
xdiff <- array(0,c(Np,Np))
for(k in 1:Np)
for(l in 1:Np)
xdiff[k,l]<-xp[k]-xp[l]
ydiff <- array(0,c(Np,Np))
for(k in 1:Np)
for(l in 1:Np)
ydiff[k,l]<-yp[k]-yp[l]
tij <- array(0,c(Np,Np))
for(k in 1:Np)
```

```
for(l in 1:Np)
tij[k,l] <- tp[k]+tp[l]

for(k in 1:Np)
for(l in 1:Np)
(
   MCM[l,k]      <-  FI(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[l,Np+k]   <- FIx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[l,2*Np+k] <- FIy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[l,3*Np+k] <- FIxx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[l,4*Np+k] <- FIxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[l,5*Np+k] <- FIyy(tij[k,l],xdiff[k,l],ydiff[k,l])
)

for(k in 1:Np)
for(l in 1:Np)
(
   MCM[Np+l,k]      <- -FIx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[Np+l,Np+k]   <- -FIxx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[Np+l,2*Np+k] <- -FIxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[Np+l,3*Np+k] <- -FIxxx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[Np+l,4*Np+k] <- -FIxxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[Np+l,5*Np+k] <- -FIxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
)
for(k in 1:Np)
for(l in 1:Np)
(
   MCM[2*Np+l,k]      <- -FIy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[2*Np+l,Np+k]   <- -FIxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[2*Np+l,2*Np+k] <- -FIyy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[2*Np+l,3*Np+k] <- -FIxxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[2*Np+l,4*Np+k] <- -FIxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[2*Np+l,5*Np+k] <- -FIyyy(tij[k,l],xdiff[k,l],ydiff[k,l])
)
for(k in 1:Np)
for(l in 1:Np)
(
   MCM[3*Np+l,k]      <- FIxx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[3*Np+l,Np+k]   <- FIxxx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[3*Np+l,2*Np+k] <- FIxxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[3*Np+l,3*Np+k] <- FIxxxx(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[3*Np+l,4*Np+k] <- FIxxxy(tij[k,l],xdiff[k,l],ydiff[k,l])
   MCM[3*Np+l,5*Np+k] <- FIxxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
)
for(k in 1:Np)
for(l in 1:Np)
(
```

```
    MCM[4*Np+l,k]       <- FIxy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[4*Np+l,Np+k]    <- FIxxy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[4*Np+l,2*Np+k] <- FIxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[4*Np+l,3*Np+k] <- FIxxxy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[4*Np+l,4*Np+k] <- FIxxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[4*Np+l,5*Np+k] <- FIxyyy(tij[k,l],xdiff[k,l],ydiff[k,l])
)
for(k in 1:Np)
for(l in 1:Np)
(
    MCM[5*Np+l,k]       <- FIyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[5*Np+l,Np+k]    <- FIxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[5*Np+l,2*Np+k] <- FIyyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[5*Np+l,3*Np+k] <- FIxxyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[5*Np+l,4*Np+k] <- FIxyyy(tij[k,l],xdiff[k,l],ydiff[k,l])
    MCM[5*Np+l,5*Np+k] <- FIyyyy(tij[k,l],xdiff[k,l],ydiff[k,l])
)
MCM[,]
det(MCM[,])


#################################################################
#Coefficient vector (Cv) calculation
#################################################################
#Feature vector
Fv <- array(0,6*Np)
for(l in 1:Np)
(
   ls <- which(Scales==tp[l])[1]
   Fv[l]       <- L[xp[l],yp[l],ls]
   Fv[Np+l]    <- Lx[xp[l],yp[l],ls]
   Fv[2*Np+l] <- Ly[xp[l],yp[l],ls]
   Fv[3*Np+l] <- Lxx[xp[l],yp[l],ls]
   Fv[4*Np+l] <- Lxy[xp[l],yp[l],ls]
   Fv[5*Np+l] <- Lyy[xp[l],yp[l],ls]
)
Fv

eps_fv <- 1e-3 #Thresholding values equal to zero
kzero <- which(abs(Fv)<eps_fv)
k_nzero <- which(abs(Fv)>=eps_fv)
Fv_nonzero <- Fv[-kzero] #Eliminating values equal to zero

MCM_nonzero<-MCM[-kzero,-kzero]

#################################################################
#Calculating inverse of MCM using SVD
```

```
MSVD<-svd(MCM_nonzero)
epsilon <- 1e-5
u0 <- MSVD$u
v0 <- MSVD$v
d0 <- MSVD$d
N0 <- length(d0)
d1 <- array(0,c(N0,N0))
for(k in 1:N0)if(d0[k]>=epsilon)d1[k,k]<-1.0/d0[k]
d1<- t(d1)
Minv_nonzero <-  t(v0%*%d1%*%t(u0))
Cv_nzero <- Fv_nonzero %*% Minv_nonzero
#Coefficient Vector
Cv <- array(0,c(1,Np*6))
Cv[1,k_nzero] <- Cv_nzero[1,]
###################################################################
#Linear reconstruction Algorithm
Arec <- array(0,c(M,N))#Reconstruction image grid
for(i in 1:M)
for(j in 1:N)
(
  st <- 0
   for(k in 1:Np)
   (
     st <- st + Cv[k]*FI(tp[k],i-xp[k],j-yp[k])
     st <- st + Cv[Np+k]*FIx(tp[k],i-xp[k],j-yp[k])
     st <- st + Cv[2*Np+k]*FIy(tp[k],i-xp[k],j-yp[k])
     st <- st + Cv[3*Np+k]*FIxx(tp[k],i-xp[k],j-yp[k])
     st <- st + Cv[4*Np+k]*FIxy(tp[k],i-xp[k],j-yp[k])
     st <- st + Cv[5*Np+k]*FIyy(tp[k],i-xp[k],j-yp[k])
   )
   Arec[i,j] <- st
)

# Absolute error:
rmse<- sqrt(sum((Arec-A)^2) / (M*N))
rmse
sum(abs(Arec-A))/(M*N)

# Reconstructed images and profiles to assess level of reconstruction

x11()
par(mfrow=c(2,3))
image(x,y, A, col=gray((0:255)/255), main = paste('Original',sep=''),xlab='',ylab='',axes=FALSE)
image(x,y, Arec, col=gray((0:255)/255), main = paste('Reconstructed',sep=''),xlab='',ylab='',axes=FALSE)

plot(Arec[,10],main='Arec[,10]',ylim=c(min(c(Arec[,10], A[,10])),max(c(Arec[,10], A[,10]))))
lines(A[,10])
```

```
plot(Arec[,20],,main='Arec[,20]',ylim=c(min(c(Arec[,20], A[,20])),max(c(Arec[,20], A[,20]))))
lines(A[,20])

plot(Arec[10,],,main='Arec[10,]',ylim=c(min(c(Arec[10,], A[10,])),max(c(Arec[10,], A[10,]))))
lines(A[10,])

plot(Arec[20,],,main='Arec[20,]',ylim=c(min(c(Arec[20,], A[20,])),max(c(Arec[20,], A[20,]))))
lines(A[20,])
```

LIST OF REFERENCES

Adelson, E. H., Anderson, C. H., Bergen, J. R., Burt, P. J., & Ogden, J. M. (1984). Pyramid methods in image processing. *RCA engineer, 29*(6), 33-41.

Akgun, T., Altunbasak, Y., & Mersereau, R. M. (2004). *Superresolution reconstruction of hyperspectral images.* Paper presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP '04).

Amolins, K., Zhang, Y., & Dare, P. (2007). Wavelet based image fusion techniques :- An introduction, review and comparison. *ISPRS Journal of Photogrammetry and Remote Sensing, 62*(4), 249-263.

Atkinson, P. M. (2004). Resolution Manipulation and Sub-Pixel Mapping. In S. M. de Jong & F. D. van der Meer (Eds.), *Remote Sensing and Image Analysis* (pp. 51-70).

Atkinson, P. M., Cutler, M. E. J., & Lewis, H. (1997). Mapping sub-pixel proportional land cover with AVHRR imagery. *International Journal of Remote Sensing, 18*(4), 917-935.

Balmachnova, E., Florack, L. M. J., Platel, B., Kanters, F. M. W., & Romeny, B. M. T. (2005). Stability of top-points in scale space. In R. Kimmel, N. Sochen & J. Weickert (Eds.), *Scale Space and Pde Methods in Computer Vision, Proceedings* (Vol. 3459, pp. 62-72).

Crowley, J. L. (2005). Perception, recognition and integration for interactive environments. *Formation et Analyse d'Images* Retrieved 28th October, 2010, from http://www-prima.imag.fr/perso/Hall/Courses/FAI05/Session9.pdf

Duits, R. (2005). *Perceptual organization in image analysis. PhD Thesis.* . Technische Universiteit Eindhoven. Department of Biomedical Engineering, The Netherlands, Eindhoven.

Farsiu, S., Robinson, D., Elad, M., & Milanfar, P. (2004). Advances and challenges in Super-Resolution. *International Journal of Imaging Systems and Technology, 14*(2), 47-57.

Fisher, P. (1997). The pixel: A snare and a delusion. *International Journal of Remote Sensing, 18*(3), 679-685.

Florack, L., & Kuijper, A. (2000). The Topological Structure of Scale-Space Images. *Journal of Mathematical Imaging and Vision, 12*, 65-79.

Florack, L., ter Haar Romeny, B. M., Koenderink, J. J., & Viergever, M. A. (1992). Scale and the differential structure of images. *Image and Vision Computing, 10*(6), 376-388.

Forberg, A. (2007). Generalization of 3D building data based on a scale-space approach. *ISPRS Journal of Photogrammetry and Remote Sensing, 62*(2), 104-111.

Gurjar, S. B., & Padmanabhan, N. (2005). Study of various resampling techniques for high-resolution remote sensing imagery. *Journal of the Indian Society of Remote Sensing, 33*(1), 113-120.

Hao, B., Li, M., & Feng, X. (2008). Wavelet iterative regularization for image restoration with varying scale parameter. *Signal Processing: Image Communication, 23*(6), 433-441.

Heller, J., & Pakzad, K. (2005). Scale-dependent adaptation of object models for road extraction. *Object Extraction for 3D City Models, Road Databases and Traffic Monitoring-Concepts, Algorithms and Evaluation*, 23–28.

Heuwold, J., Pakzad, K., & Heipke, C. (2007). Scale behaviour prediction of image analysis models for 2D landscape objects. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, 36*(3/W49A), 43–50.

Hossny, M., Nahavandi, S., & Creighton, D. (2007, 23-27 June). *A Quadtree Driven Image Fusion Quality Assessment.* Paper presented at the 5th IEEE International Conference on Industrial Informatics.

Hughes, C. G., & Ramsey, M. S. (2010). Super-resolution of THEMIS thermal infrared data: Compositional relationships of surface units below the 100 meter scale on Mars. *Icarus, 208*(2), 704-720.

Iijima, T. (1962). Basic theory on normalization of a pattern (in case of typical one-dimensional pattern). *Bulletin of Electrical Laboratory, 26*, 368-388.

Janssen, B., Kanters, F., Duits, R., Florack, L., & Romeny, B. (2006). A linear image reconstruction framework based on Sobolev type inner products. *International Journal of Computer Vision, 70*(3), 231-240.

Jong, S., Meer, F., & Clevers, J. (2004). Basics of Remote Sensing. *Remote Sensing Image Analysis: Including The Spatial Domain*, 1-15.

Kanters, F. (2007). *Towards object-based image editing. Phd Thesis.* Eindhoven University of Technology, The Netherlands, Eindhoven.

Kanters, F., & Florack, L. (2003a). Software for Visualizing MSSTs from 2D images (Version 1.0). Eindhoven, Netherlands: Information Society Technologies.

Kanters, F., Florack, L., Platel, B., & ter Haar Romeny, B. (2003b). Image Reconstruction from Multiscale Critical Points. In L. Griffin & M. Lillholm (Eds.), *Scale Space Methods in Computer Vision* (Vol. 2695, pp. 464-478): Springer Berlin / Heidelberg.

Kanters, F., Lillholm, M., Duits, R., Janssen, B., Platel, B., Florack, L., & Romeny, B. T. (2005). Image Reconstruction from Multiscale Top Points. In R. Kimmel, N. Sochen & J. Weickert (Eds.), *Scale Space and Pde Methods in Computer Vision, Proceedings,* (Vol. 3459, pp. 431-442): Springer Berlin / Heidelberg.

Kanters, F., Platel, B., Florack, L., & ter Haar Romeny, B. (2003c). Content Based Image Retrieval Using Multiscale Top Points. In L. D. Griffin & M. Lillholm (Eds.), *Scale Space Methods in Computer Vision, Proceedings,* (Vol. 2695, pp. 33-43): Springer Berlin / Heidelberg.

Koenderink, J. (1984). The structure of images. *Biological Cybernetics, 50*(5), 363-370.

Kuijper, A. (2002). *The deep structure of Gaussian scale space images. Phd Thesis.*, The University of Utrecht, Utrecht, Netherlands.

Kuijper, A., & Florack, L. (2001). *The application of catastrophe theory to medical image analysis*: Technical Report UU-CS-2001-23, Dept. of Computer Science, Utrecht University, Netherlands.

Kuijper, A., & Florack, L. (2002). Understanding and Modeling the evolution of critical points under Gaussian blurring. In A. Heyden, G. Sparr, M. Nielsen & P. Johansen (Eds.), *Computer Vison - ECCV 2002, Pt 1* (Vol. 2350, pp. 143-157): Springer-Verlag Berlin

Kuijper, A., Florack, L. M. J., & Viergever, M. A. (2003). Scale space hierarchy. *Journal of Mathematical Imaging and Vision, 18*(2), 169-189.

Lehmann, T. M., Gonner, C., & Spitzer, K. (1999). Survey: Interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging, 18*(11), 1049-1075.

Lindeberg, T. (1990). Scale-space for discrete signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 12*(3), 234-254.

Lindeberg, T. (1994). *Scale Space Theory in Computer Vision*. Dordrecht: The Kluwer International Series.

Lindeberg, T. (1998). Feature detection with automatic scale selection. *International Journal of Computer Vision, 30*(2), 79-116.

Marr, D., & Hildreth, E. (1980). Theory of edge detection. *Proceedings of the Royal Society of London. Series B, Biological Sciences, 207*(1167), 187-217.

Mather, P. M. (2004). *Computer processing of remotely-sensed images: An introduction* (Third edition ed.). Chichester etc.: Wiley & Sons.

Nielsen, M., & Lillholm, M. (2001). What do features tell about images? *Scale-Space and Morphology in Computer Vision*, 39-50.

Nishihara, H. (1984). Prism: A practical real-time imaging stereo matcher. *Journal of Optical Engineering, 23*(5), 536-545.

Romeny, B. M. t. H. (2002). *Front-End Vision and Multi-Scale Image Analysis* (XVII ed.): Kluwer Academic Publishers, Dordrecht.

Shan, Q., Li, Z. R., Jia, J. Y., & Tang, C. K. (2008). Fast Image/Video Upsampling. *Acm Transactions on Graphics, 27*(5).

Song, C. H. (2005). Spectral mixture analysis for subpixel vegetation fractions in the urban environment: How to incorporate endmember variability? *Remote Sensing of Environment, 95*(2), 248-263.

Stefanidis, A., & Schenk, T. (1993). On the Application of Scale Space Techniques in Digital Photogrammetry. *International Archives of Photogrammetry and Remote Sensing, 29 Part B3 comm. 3.*, 586-590.

Tolpekin, V. A., & Stein, A. (2009). Quantification of the Effects of Land-Cover-Class Spectral Separability on the Accuracy of Markov-Random-Field-Based Superresolution Mapping. *Geoscience and Remote Sensing, IEEE Transactions on, 47*(9), 3283-3297.

Tso, B., & Mather, P. M. (2001). *Classification methods for remotely sensed data.* London: Taylor & Francis.

Verhoeye, J., & De Wulf, R. (2002). Land cover mapping at sub-pixel scales using linear optimization techniques. *Remote Sensing of Environment, 79*(1), 96-104.

Vu, T. T., Yamazaki, F., & Matsuoka, M. (2009). Multi-scale solution for building extraction from LiDAR and image data. *International Journal of Applied Earth Observation and Geoinformation, 11*(4), 281-289.

Weickert, J. (1998). *Anisotropic diffusion in image processing. Phd Thesis*: Teubner Verlag, Stuttgart.

Weickert, J., Ishikawa, S., & Imiya, A. (1999). Linear scale-space has first been proposed in Japan. *Journal of Mathematical Imaging and Vision, 10*(3), 237-252.

Wolfram Research Inc. (2010). Mathematica (Version 8.0). Champaign, Illinois: Wolfram Research, Inc.