IMAGE SEGMENTATION USING IHS SPACE AND OBJECT-BASED ANALYSIS

Carlos Frederico de Sá Volotão

Doctorate Thesis at Graduate Course in Applied Computing, guided by Drs. Luciano Vieira Dutra, and Rafael Duarte Coelho dos Santos, approved in February 20, 2013.

URL of the original document:
<http://urlib.net/8JMKD3MGP7W/3DDRPPL>

INPE

São José dos Campos

2013

IMAGE SEGMENTATION USING IHS SPACE AND OBJECT-BASED ANALYSIS

Carlos Frederico de Sá Volotão

Doctorate Thesis at Graduate Course in Applied Computing, guided by Drs. Luciano Vieira Dutra, and Rafael Duarte Coelho dos Santos, approved in February 20, 2013.

INPE

São José dos Campos

2013

Aprovado (a) pela Banca Examinadora em cumprimento ao requisito exigido para obtenção do Título de *Doutor(a)* em

Computação Aplicada

Dr.   Solon Venâncio de Carvalho

Presidente / INPE / SJCampos - SP

Dr.   Luciano Vieira Dutra
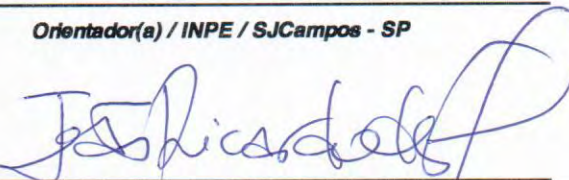
Orientador(a) / INPE / SJCampos - SP

Dr.   Rafael Duarte Coelho dos Santos

Orientador(a) / INPE / SJCampos - SP

Dr.   João Ricardo de Freitas Oliveira

Membro da Banca / INPE / SJCampos - SP

Dr.   Antônio Henrique Correia

Convidado(a) / DSG/EB / Brasília - DF

Dr.   Vagner Braga Nunes Coelho

Convidado(a) / IME/EB / Rio de Janeiro - RJ

Este trabalho foi aprovado por:

( ) maioria simples

unanimidade

Aluno (a): *Carlos Frederico de Sá Volotão*

São José dos Campos, 20 de Fevereiro de 2013

"The pot he was shaping from the clay
was marred in his hands;
so the potter formed it into another pot,
shaping it as seemed best to him."

Jeremiah 18:4

I dedicate this work to my wife, Vivian,
and to my children, Eric and Stephanie.

ACKNOWLEDGEMENTS

ABSTRACT

Segmentation is an important procedure in remote sensing image analysis, which divides an image into parts with uniform properties and changes the smallest unit of an image from pixel to segment. Some factors produce undesired results in the segmentation of remote sensing imagery and one of such factors is due to illumination: the occurrence of shadows. Lighting affects image segmentation because variations on scene lighting modifies the pixel values in all spectral bands. The concentration of intensity on the sensed signal in one channel produces two other channels, hue and saturation, where the effects of glare, shadows and gradients are minimized. The hue is being used to identify objects that are distinguishable by this attribute, and the same principle of detection is being extended to multi-band images, but it is not suitable for all cases. To improve the process it is being presented a way to produce a saturation-weighted synthetic hue channel for multispectral imagery. The central idea behind this object-based approach is the ability to evaluate and change any segment before finishing the segmentation process. A turning function is a representation of polygons by angles and lengths and it may be analyzed and modified as necessary. Every segment identified as foreground undergoes to analysis and the corresponding segment is subject to changes. To obtain the turning function, the segment is first identified as a blob and then converted into a Freeman's chain code format. The use of indexes is helpful to categorize shapes and some metrics are being presented. The algorithm is implemented in IDL language and it have two modes: unsupervised and supervised. The unsupervised uses a region growing algorithm with random seeds. The supervised consists on the manual indication of a small area of the object. Besides the enhancements on the algorithms and the proposal of a object-based approach for a chosen segmentation technique, this thesis also proposes a feasible way to make segmentation based on phase extracted from multispectral imagery using any segmentation software in addition to creating a synthetic hue band from the combinations of color compositions of the original imagery.

SEGMENTAÇÃO DE IMAGENS USANDO ESPAÇO IHS E ANÁLISE BASEADA EM OBJETO

RESUMO

A segmentação é um procedimento importante na análise de imagens de sensoriamento remoto, que divide uma imagem em partes com propriedades uniformes e muda a menor unidade de uma imagem de pixel para segmento. Alguns fatores produzem resultados indesejados na segmentação de imagens de sensoriamento remoto e um desses fatores é a existência de sombras na cena. A iluminação afeta a segmentação de imagens porque as variações na iluminação da cena modificam os valores de pixel em todas as bandas espectrais. A concentração das informações de intensidade do sinal detectado em um canal minimiza os efeitos do brilho, sombras e degradês nos outros canais (i.e., matiz e saturação). A matiz está sendo usada para identificar objetos que se distinguem por esse atributo, e o mesmo princípio de detecção está sendo estendido para imagens de múltiplas bandas, embora não seja indicado em todos os casos. Para melhorar o processo é apresentado um modo de produzir um canal sintético de matiz fazendo-se a média de múltiplos canais de matizes ponderados pela saturação, correspondentes a bandas multiespectrais de uma imagem. A idéia central por trás dessa abordagem baseada em objetos é a capacidade de avaliar e alterar qualquer segmento antes da conclusão do processo de segmentação. Uma função de desvios (*turning function*) é um modo de representação de polígonos pelos ângulos e comprimentos, que pode ser analisada e modificada conforme necessário. Cada segmento da imagem pertencente ao primeiro plano é submetido a análise e o segmento correspondente fica sujeito a alterações, quando necessário. Para se obter a função de desvios, primeiro o segmento é representado de modo binário (i.e., como um *blob*) e, posteriormente, convertido para o código da cadeia de Freeman. O uso de índices é útil para categorizar formas e algumas métricas são apresentadas. Os algoritmos são implementados em linguagem IDL em dois modos: supervisionado e não-supervisionado. O primeiro consiste na indicação na imagem de uma pequena área do objeto. O segundo usa um algoritmo de crescimento de região com sementes aleatórias. Além da revisão do algoritmo original, ampliando suas capacidades, e a proposta de uma abordagem baseada em objeto para a técnica de segmentação escolhida, esta tese propõe um modo de fazer a segmentação em função do matiz extraído de imagens multiespectrais, utilizando um aplicativo de segmentação, e também cria uma banda sintética de matiz a partir das combinações das composições coloridas da imagem original.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

GCP   –   Ground Control Point
GIS   –   Geographic Information System
HSI   –   Hue-Saturation-Intensity (color system)
HSV   –   Hue-Saturation-Value (color system)
HVS   –   Human Visual System
IHS   –   Intensity-Hue-Saturation (color system)
INPE   –   Instituto Nacional de Pesquisas Espaciais
IA   –   Image Analysis
ID   –   Identifier (i.e., the numeric code representing a segment)
IDL   –   Interactive Data Language, a programming language
OBIA   –   Object-Based Image Analysis
PCA   –   Principal Component Analysis
ROI   –   Region of Interest (i.e., image samples)
SegmenTHIS   –   Name of the segmentation prototype (software) based on hue, intensity and saturation information.
SPD   –   Spectral Power Distribution
TF   –   Turning Function

LIST OF SYMBOLS

θ	–	vector of tangent angles of the turning function
$\theta(s)$	–	vector of tangent angles of the turning function, af function of
s	–	vector of segments of the turning function
θ_{Equiv}	–	the equivalent tangent angle of a turning function interval
s_{Equiv}	–	the equivalent segment of a turning function interval
(x, y)	–	a point in an image
f	–	image (function)
f_λ	–	function dependant of the wavelength
$\bar{f}$	–	mean of an image f
var	–	variance of an image f
σ_s	–	the sample standard deviation of an image f
g	–	a segmentation
g^{-1}	–	the inverse of the segmentation g
g_{sin}	–	segmentation of the sine channel
g_{cos}	–	segmentation of the cosine channel
ϕ	–	hue channel information
L	–	radiance of a target
R, G, B	–	red, green, blue channels
t	–	time
p	–	polarization factor
r	–	spectral reflectance of an object
λ	–	electromagnetic wavelength
M	–	spectral radiant emittance of a blackbody

CONTENTS

1 INTRODUCTION

1.1 Contextualization

Segmentation is an important procedure in remote sensing image analysis, which divides an image into parts with uniform properties, changing the smallest unit of an image from pixel to segment. It is important for remote sensing and also for the delineation of anatomical structures, handwritten character recognition, object tracking, security scanner systems and many other applications (PARK, 2003; LANDSMEER, 2007).

(a) Original image   (b) Segmentation boundaries p=(25,500)
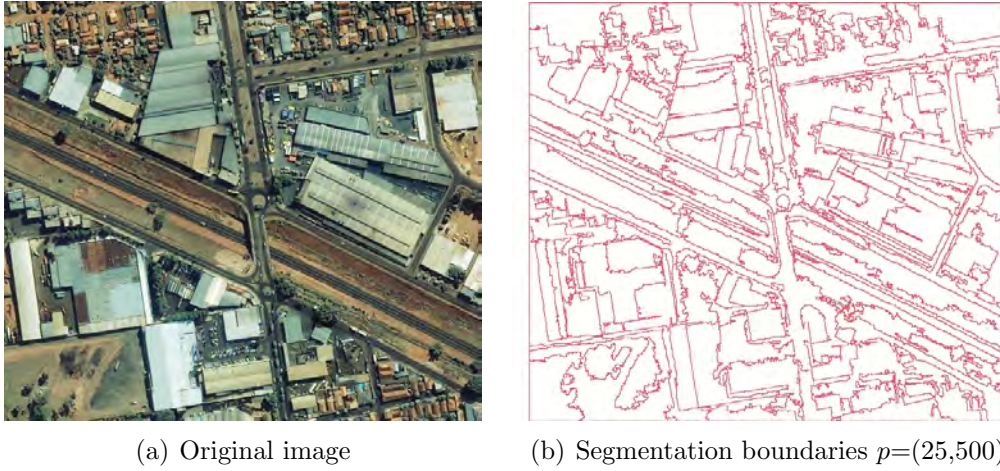
Figure 1.1 - Ikonos image, São José dos Campos, SP, Brazil.

It is not a simple process and therefore multiple techniques arise with different approaches to achieve segmentation tailored for the type of image, application and target. Segmentation produce edges or regions, but perfection on these are not always required for an application (BALLARD; BROWN, 1982).

To exemplify the complexity of segmentation, Figure 1.1 shows an example of an satellite urban image and a segmentation of the corresponding image using Spring [1], with its region-growing method. Region-growing algorithms typically require user-supplied control parameters [2] (ESPINDOLA et al., 2006). The example of Figure 1.1(b) shows the boundaries of a segmentation, using the parameters p=(25,500), which

---

[1] Spring is a freeware software developed by the National Institute for Space Research (INPE) with image engineering (ZHANG, 2009) routines, a GIS interface and tools and includes some vector capabilities.

[2] Spring's parameters are p=(similarity threshold, area threshold)

means similarity threshold of 25 and area threshold of 500. The result is strongly dependent on the resolution and the target. This is a way of presenting segmentation results, but other ways are shown below in other examples in this text.

Changing the parameters implies in a different segmentation and usually some tests are necessary in any segmentation software to achieve the expected results: an output with the desirable targets (objects or terrain features) detected individually. This level of information is exemplified by Figure 1.1(b), showing the boundaries of the expected result based on user needs.
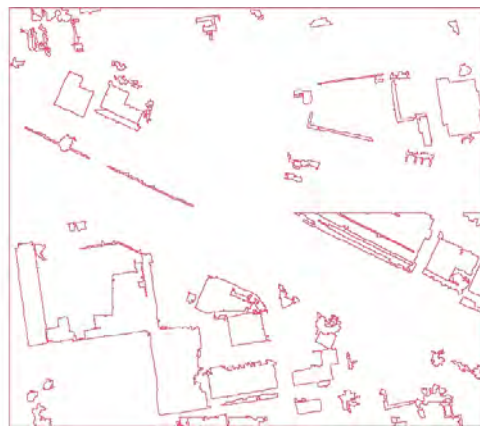
Figure 1.2 - Under-segmentation p=(80,80) of Figure 1.1(a).

Figure 1.2 represent an under-segmentation output while Figure 1.3(a) and Figure 1.3(b) are over-segmentations of the same image. A common segmentation technique is to use over-segmentation as a pixel grouping process, producing super-pixels. The next step is a merging of segments to compose the desirable objects.

Image segmentation plays a very important role in remote sensing applications because it depicts new possibilities of image analysis. Pixel-based methods are not appropriate to extract objects from the spectral information of images with high spatial resolution.

Image classification is a basic process that assigns semantic information, e.g. labels, to pixels and regions. When available, ground truths may be used in the process. Segmentation approach can be used to include the context information of pixels and thus giving an object-level classification, despite the existence of pixels with different values in the same image area segment. The classification of segments produce a better looking result.

(a) Over-segmentation p=(5,200) (b) Over-segmentation p=(5,10)
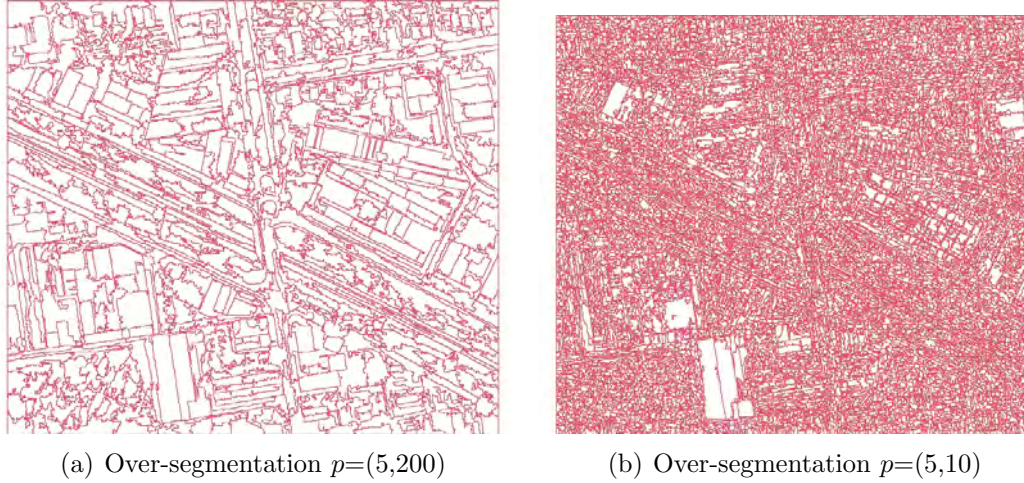
Figure 1.3 - Over-segmentation of Figure 1.1(a).

Grouping mechanisms that organize the scene into meaningful units are a significant step toward image understanding (BALLARD; BROWN, 1982). Scene understanding is a high-level step and uses objects and its context to interpret what is happening on the real world in a scene.

Geographic databases are being widely used with new technology applications (e.g., GPS devices, military applications, aerial and terrestrial navigation systems and so on) and the update of these databases depends on the identification of objects. The complexity of the urban environment motivated the development of the object-based approach.

A broad field is opened with the emergence of Object-Based Image Analysis (OBIA) concept. OBIA propounds that we can take advantage on moving from handling pixels to manipulate regions. The object-based proposition is the treatment of the produced segments as objects so that it is possible to expand the possibilities of treating the pixels. Acting together, it is possible to think differently, and this is opening new possibilities not conceived previously.

OBIA is a new approach, simple in theory but complex to be implemented. Images can be divided in small homogeneous regions first, based on spectral or textural information, and then group these regions according to spectral, geometric or contextual criteria. Segmentation is so a crucial step, because it yields the separation of the objects or its parts into images. The examples of segmentation shown in Sections 1.1, 1.2 and 1.3 illustrates the the boundaries of some possible variations by changing segmentation parameters. The user is expected to have knowledge of the

3

objects of interest and to make several tests in order to be able to chose the best parameters to extract the objects (SEBARI; HE, 2007; JENSEN, 2004).

1.2 Motivation

Considering remote sensing image processing, the reasons for segmentation's undesired results are based on a number of issues. Some phenomena appear in the image (Figure 1.4).
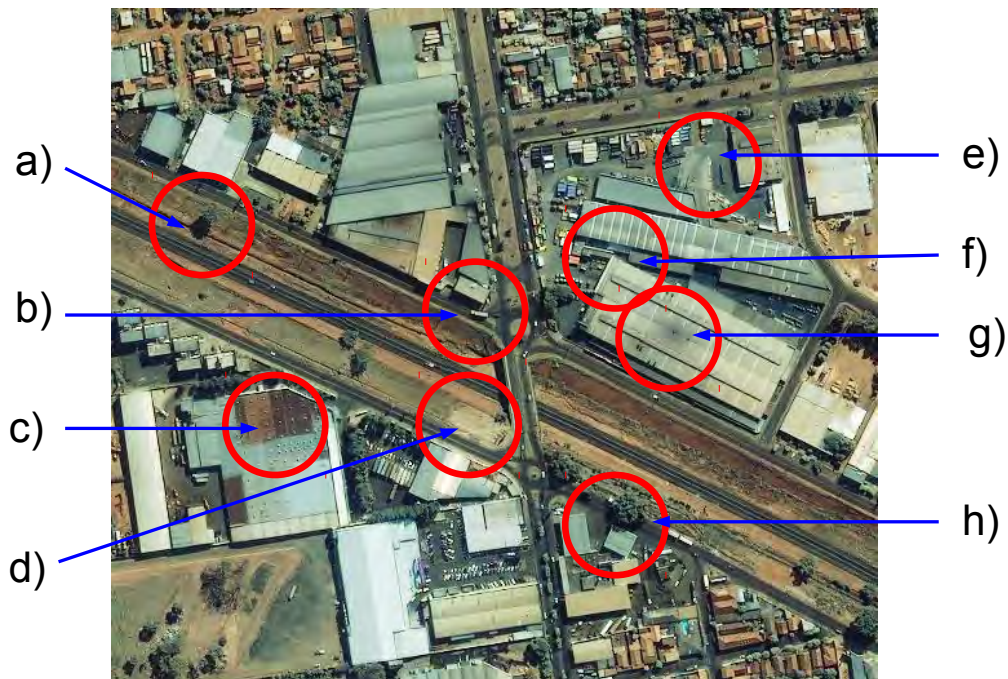
Figure 1.4 - Physical phenomena that occur in remotely sensed imagery and hamper obtaining the expected results in segmentation.

Some examples of problems that can affect remote sensing images and may not be desirable to be represented on segmented images, depending on the application, are exemplified on Figure 1.4:

a) shadow of a tree is a dark area with a complex and rounded shape;

b) vehicles are mobile objects with regular shapes (usually to be discarded);

c) different materials or shapes compounding objects by aging;

d) irregularities in targets with no relevance (soil deterioration);

e) changes in targets due to aging (tire marks on the asphalt);

f) occlusion due to imaging geometry (part of the street is missing);

g) chimney gradient effect on a roof;

h) shadow of the tree over the square roof.

The segment we are concerned in this context is the shape of the targets that can be considered as objects of interest. These objects are present in high and very high resolution images of urban area. In this sense we define objects as segments related to the outlines of the objects on real world.

The recovery of shapes in remote sensing imagery suffer from additive errors. Thompson et al. (1996) tackles the recovery of geometric models from sensed data, and present the measure of the quality of a reconstructed model as the difference between actual geometry and the one of the reconstructed model. In remote sensing, usually the "truth" for urban area shapes is a geometrically regular shape, except a very common target: the trees.

Some factors can be pointed out and for the same kind of objects there are variations in segment shape and region due to:

a) built variations,

b) changes over time due to aging effect,

c) occlusion by objects,

d) clouds and shadows,

e) uneven illumination,

f) sensing error due to pixel geometric and radiometric resolution,

g) atmospheric effects,

h) sensor properties and

i) imaging geometry.

The sum of all these factors implies the need to rectify the resulting shape. As an example, it is undesirable for a rectangular object like a roof to be represented by

an irregular polygon as a result of both shadow and partial occlusion by a crown canopy. In some cases it may be possible to identify occlusion based on the resulting shape (BEUSMANS et al., 1987).

Several undesired aspects in segmentation existing in almost all remotely sensed imagery are related to illumination. The occurrence of shadows and uneven illumination affects heavily image segmentation and variations on scene lighting modifies the pixel values in all spectral bands.

Physical aspects related to the imaging scene produce undesirable issues in segmentation. Problems with shape include partial match, when only part of the object is identified, occlusion split, holes, disconnected region, scale, noises, speckle, spikes, and change in geometry by projection, deformation, distortion and object intrinsic change.

Figure 1.5 - Detail on segmentation problems of Figure 1.4.

Figure 1.5 shows what happens when remotely sensed images are segmented (Figure 1.4). These issues makes difficult to get a perfect segmentation in one single

segmentation process. This is why usually it is necessary to use at least two stages, the first making a over-segmentation and then merging regions depending on the objects in scene.

The variations in solar lighting specially in remote sensing urban imagery due to object shapes or other objects blocking sunlight leads usually to segmentation issues of illumination. To minimize this effect and to enhance an existing segmentation algorithm we assembled a shape processing module inside the hue-based classifier (VOLOTAO et al., 2011). The hue channel obtained by a perceptual transformation can be used to reduce significantly the effect of illumination. Hue channel quality is very sensitive to original image coding used to compress data, as it can degrade the quality of hue information (MIYAHARA, 1988).

Hue segmentation of multispectral images (SOUTO, 2000) can discriminate colored terrain targets such as pools, cars, roofs, roads, trees, gardens and several other urban features and objects, reducing or eliminating the illumination effects due to the electromagnetic radiation variations over the targets.

To enhance the details on each foreground segments (i.e., the objects), if we had a method of segmentation associated with shape handling capabilities, we possibly could better solve minor faults detected on original segments. A shape module can help to demonstrate the feasibility of this approach and could open a new field on shape analysis coupled with a segmentation method.

The use of hue segmentation with a shape manipulation tool, based in a turning function representation, is being presented as a possible solution to reduce shadows, glares and gradient lighting effect on some remote sensing urban scenes and allows producing closer segments to given object shape properties. The basic concept of object as a set of pixels is the grounding of an emerging field of image analysis. Advanced techniques emerge from this naive set assumption, resulting in powerful outcomes.

1.3 Related work

1.3.1 Hue segmentation

According to Gong and Sakauchi (1995), the separation of the chromatic from the luminant component of color is an effective way to detect constant colored regions under nonuniform illumination conditions, such as shade, highlight and sharp contrast. Uneven illumination conditions limit the results of segmentation process that

uses both luminant and chromatic components of color. This can be fixed, in most cases, only by a higher level of interpretation mechanisms, applied to give meaning to the initial results.

Souto (2000) implemented segmentation algorithms and classification of hue images in regions where there are differences in luminosity of forest and urban area. The implemented algorithm was mostly effective in these two cases. A method was implemented to get the hue from N spectral bands (with $N \geq 3$), but the experiments showed that increasing the number of bands did not reflect, always, a better classification. Classification was compared to verify if there were significant differences between the classification of hue from three, four and five bands. The best performance of a given target (e.g., urban area) can be achieved by a determined number of bands (e.g., in the study of Souto (2000) the urban targets existing in mid/low-resolution imagery provided a better performance using exactly four bands).

The result of the classifications using hue, obtained by (SOUTO, 2000), compared with Spring's ISOSEG – an unsupervised clustering method used after segmentation (BINS et al., 1993) – was favorable for the hue, except when using ISOSEG with five bands. Dark areas produced an unstable value and meaningless, making necessary to mask these regions and create a new class of segments. The main points observed were:

a) the order of the spectral bands to provide hue is irrelevant;

b) adding a new band, depending on the case, can improve or worsen the outcome;

c) the hue channel became unstable in areas of low brightness, but the creation of a new class solved the problem;

d) the hue method performed better than ISOSEG in most cases.

Souto's work tested forest and urban area from LANDSAT/TM with $30m$, and the segmenter was developed in IDL. Because of the results, that algorithm served as basis for the present thesis.

1.3.2 Object-based image segmentation

Korting (2008) developed a resegmentation methodology focusing regular objects in high-resolution images, aiming to improve an existing segmentation. The approach starts with classified over-segmented images and uses spectral and shape

attributes and information from segment adjacency. The focus was the extraction of rectangular-shaped urban objects (e.g. roofs, swimming pools, blocks) (KORTING et al., 2008; KORTING et al., 2010; KORTING et al., 2011).

Uba (2009) proposed a methodology for regular objects estimation applied to high resolution imagery, based on rectangular shapes, using template matching. The proposed method uses multiple spectral bands and a previous over-segmentation, from where it extracts a set of polygons. A classification is previously performed and proposed a template matching technique to process and redefine the polygons of the interesting classes, resulting in a new set of polygons that represents the urban objects on image in a manner that is visually better.

1.3.3 Turning function

To refine the segmentation we make individual changes on each segment and, one by one, the segments are analyzed and can be modified or may give shape information to the segmentation. This approach can be extended in future to change multiple segments simultaneously, attaining the whole image.

To deal with the shapes of each segment, Guliato et al. (2008) iteratively uses a turning function as an indicator of the sizes and angles among the segments. Other works presents various methods of polygonal approximation by turning functions (RANGAYYAN et al., 2008; CARVALHO et al., 2007; PINHEIRO; GHANBARI, 2004). The results indicated the use of turning functions as a flexible modeler because it provides a new way to represent the shapes, with no loss, where the change in representation could give some new shape analysis tools.

1.4 Objective and justification

1.4.1 Objective

The proposal of this thesis is to present a different approach based on the concept of region segment as being an object during segmentation allowing further developments. The objective of the research was to develop a methodology based on a working segmentation algorithm, making possible to produce segments more consistently with the expected results, to extract other classes of information while segmenting and putting more robust and powerful tools in the hands of the image analyst.

This work introduces the capability to proceed some shape analysis operations, by

turning function, inside a hue segmentation algorithm. This gives to the segmentation the ability to produce segments whose shapes can be refined by rules.

The images must represent colored objects surrounded by a second colored background or object because of the hue extraction mechanism. It is given a focus on urban area objects, where rules on contours are plentiful and the amount of detail existing in images of very high resolution makes the shapes complex.

1.4.2 Hypothesis

The present thesis started on the hypothesis of being possible to development a new segmenter that would be able to deal with segments, suggesting an object-based segmentation approach. The use of segment shape properties inside the algorithm of segmentation defines an object-based operation. To deepen the concept of object, first the very notion of object and its properties and context is required.

1.4.3 Justification

To achieve the objective, several distinct fields of research had to be studied, and the studies included the following aspects:

a) to reduce the issues caused by illumination by the use of the IHS color system, instead of the original spectral bands, to segment high resolution remote sensing urban images;

b) to improve the performance of the hue-based segmentation algorithm developed by Souto (2000);

c) to allow the analysis of the shape by a turning function approach;

d) to propose an object-based image segmentation approach by the use of shape analysis inside the segmentation process.

Contiguous regions are expected to be the objects. An object-based (or object-oriented) approach depends on the use of grouping of pixels. So, the segmentation method starts an object-based image analysis (OBIA) procedure, but is not object-based because it only creates – and does not take advantage of the object nature of the segments being created. The basic elements of OBIA methods are the segments (contiguous regions) but only when considered objects (BAATZ et al., 2001; LI; GONG, 2006). By this definition a meaningless segment would not be considered an object, and therefore is being considered a background segment.

1.4.4 Contributions of the present work

To achieve best results in segmentation and to meet the proposed objectives, this thesis aimed to present a feasible approach based on objects. The following are contributions of this thesis:

a) the proposal of an object-based image segmentation methodology;

b) enhancements on a hue segmentation algorithm;

c) the presentation of new ways to make hue segmentation based on multi-spectral remote sensing imagery;

d) the proposal of a shape analysis approach by turning function.

1.5 Organization

Chapter two presents some image processing and analysis concepts used in this thesis. It is divided in four sections: the first three presenting the overview of the chapter, the segmentation concept, the creation of objects and the use of these objects by high cognitive level operations. The fourth section presents the idea of image understanding, a very important subject that depends on the identification of the shapes, so it is one reason to the segmentation to be object-based.

Chapter three presents the theoretical methodology developed in this thesis and is the core of the research. It presents a brief description of the chapter, then the segmentation of color images section, devoted to multispectral process, IHS classical transforms and the synthetic hue from multispectral bands proposed, a turning function section, explaining the model chosen to represent the shapes, and, lastly, the description of the developed algorithm.

Chapter four presents results of the implemented algorithm. It is based on the methodology presented in chapter 3 and presents examples of hue segmentation, turning functions and what can be done with turning functions, and the last section shows the features available in the algorithm. Some image are shown to demonstrate the methods presented, and a description of the main aspects are given. Several aspects involving the method and its implementation is addressed in this chapter.

The conclusion, in chapter five, points the uses, the limitations and the contribution of this thesis, presenting the expected and achieved results of the proposed method-

ology and briefly highlighting the contributions, as well as the possibilities that may arise from this and future works.

2 IMAGE PROCESSING AND ANALYSIS CONCEPTS

2.1 Introduction

Image segmentation is the grouping of pixels, according to some criteria, where each group is formed by adjacent pixels (FU; MUI, 1981; HARALICK; SHAPIRO, 1985; IKONOMATAKIS et al., 1997). The adjacency is defined by the method, and there are primarily four types of segmentation techniques (SHIH; CHENG, 2005):

a) thresholding – where clusters in the histogram relate to background or objects of interest;

b) edge-based – for a region being characterized by abrupt changes in pixel properties (e.g., intensity, color, texture)

c) region-based – considering that adjacent pixels within the same region have similar values (e.g., intensity, color, texture); and

d) hybrid techniques – combine boundary detection and region growing together.

Digital images are the input, so pixels are the starting point to image engineering. Segments alone are not useful, unless they are an intermediate step for some other application. If some segments can be related to objects, a wide field of applications becomes available, because this kind of identification in a scene allows advanced analysis. The object-based techniques are being greatly developed and many new conferences, addressing object-based image analysis and understanding subjects, arise annually.

This section present this context to make clear the purpose and context of the theme of this proposal, and to demonstrate the importance of the subject.

2.2 From Pixels to Segments

In segmentation, pixels are analyzed in its context, with the return of a set of image segments. There are many approaches to proceed this grouping process (see Vantaram and Saber (2012) for a survey of methods), but neither is perfect. The main reason is that the targets usually are not so well behaved that all identifiable objects in the image can be detached from each other, for any application, uniquely and completely.

The large number of existing segmentation techniques are commonly divided into methods:

   a) edge-based techniques,

   b) region-based approach.

Approaches (a) and (b) explore spatial conformation, by detection of discontinuities or homogeneities, respectively. The region-growing algorithm, a region-based method, was used in this thesis for multispectral imagery as input.

Automatic segmentation is a difficult task because there are many combinations of conditions and objectives. Considering only remote sensing, important characteristics – to be obtained by segmentation of the images – vary, depending on the type of sensor, the type of target, the scene and the application.

The next sections describes the main problem that must be considered in segmentation of very high resolution color images.

2.2.1 Problems with RGB color image

Color composite RGB is the default color space used for image capturing and display (MAKADIA et al., 2010) but small changes in the illumination modifies the pixel color (TEIXIDÓ et al., 2012). The illumination of remotely sensed landscapes is an important factor to the proper detection of the target.

In many cases the RGB color space does not produce a satisfactory segmentation, because some targets have similar properties in this space, e.g., vegetation and ground pixels are difficult to discriminate because inconsistent field illumination influences the color information of field images (JEON et al., 2011). The close distribution of RGB values causes a dependency on illumination, and this makes each multispectral band to have very similar properties in the RGB space (RUIZ-RUIZ et al., 2009).

Each three multispectral bands can be associated to channels R (red), G (green) and B (blue), for a color composite. This way the device (e.g., screen or printer) can show a colored image.

The radiance L is presented in equation (2.1) and depends on spectral value of electromagnetic radiation (MOIK, 1980). For each position x,y, considering the time t, a polarization factor p, then the spectral reflectance of the object r is a function

$r = r(x, y, \lambda, t, p)$. The solar incident illumination, formally called spectral irradiance on the object is $i(x, y, \lambda, t)$, and the spectral radiant emittance of a blackbody is $M(\lambda)$. Based on this equation, each region of spectrum have different reflection and emission components.

$$L(x, y, \lambda, t, p) = \underbrace{(1 - r(x, y, \lambda, t, p)) \cdot M(\lambda)}_{\text{emission from the surface}} + \underbrace{r(x, y, \lambda, t, p) \cdot i(x, y, \lambda, t)}_{\text{energy reflected over the surface}} \qquad (2.1)$$

Let's simplify the equation and consider each spectral band a two-dimensional function $f_\lambda(x, y)$, despite the lambda and time dependence. Saying simply, it depends on the intensity $i_\lambda(x, y)$ of electromagnetic radiation from the source (usually the sun) and the reflectance $r_\lambda(x, y)$ of the target. So, a remote sensed image is the function f_λ as presented by equation (2.2), where $0 \leq r_\lambda(x, y) \leq 1$ and $0 \leq i_\lambda(x, y)$ (GONZALEZ; WOODS, 2010).

$$f_\lambda(x, y) = i_\lambda(x, y) \cdot r_\lambda(x, y) \qquad (2.2)$$

If i_λ was a constant function the image would represent only r_λ values, depending on the target exclusively. But unfortunately the illumination depends on several factors, including:

a) incident angle of the source;

b) angle of the target (depending on several factors);

c) atmosferic effects (e.g., scattering and abortion);

d) occlusion by other target (e.g., building);

e) solar irradiance at the top of the atmosphere;

f) the range of the wavelength (λ).

2.2.2 Other color spaces

Brightness and lightness are psychological quantities, defined by the perception of color. Brightness refers both to an absolute level of perception on the target and a comparison level in light intensity related to a second target, and so, it is a visual sensation. Lightness is the relative brightness of a target while compared to a similarly illuminated highly transmitting area. Intensity, however, is the physical measure of the energy.

Intensity variation adds a large number of variables not dependent on the target itself. These factors are related to the source, the neighborhood, the weather, the terrain slope, other sources of lighting etc. Other color spaces can be used to minimize these variations in intensity. A color space (also color model or color system) are models oriented towards hardware or software and allow the specification of colors in a standard way. Perceptual color spaces are color spaces that works in a way the human can understand, making easier to understand the color being produced.

Forsyth and Ponce (2011) defines hue as being the property of a color that varies in passing from red to green; define saturation as being the property of a color that varies in passing from red to pink; and defines brightness, lightness and value as being the property that varies in passing from black to white.

Humans describe colors in terms of hue, saturation and brightness. IHS, also known as HSI or HSV, is the ideal tool for developing image processing algorithms and is natural and intuitive to humans. IHS space can be represented by a spherical space (JENSEN, 2004),

The use of an alternative color space (instead of the most commonly used RGB) is a way of seeking an alternative to some problems we find in remote sensing imagery. There are other reasons to use alternative color spaces in image processing, as in merging distinct spatial resolution on one product, as the well-known image fusion. This is done in four steps: RGB to IHS transform, contrast manipulation, substitution of the intensity channel with the higher-spatial resolution one, and, finally, returning from IHS to RGB (JENSEN, 2004).

Color spaces can be divided into two groups: created by linear transform from RGB and created by nonlinear transform from RGB. Color spaces provide color invariants (e.g. invariance to illumination), color histogram and color texture that can be powerful information.

Intensity can be a key property in object detection (SHIH; LIU, 2005; MARTINKAUPPI et al., 2001), but in other cases, the negative effect of shadows and highlights, producing false classification results, can be greatly reduced by the use of an alternative color space (KUMAR et al., 2002).

2.2.3 Problems with very high resolution color images

A simple example, using Figure 2.1(a), can show how difficult is to have an algorithm that satisfies multiple users' goals. In the figure, distinct applications may ask for

different segmentation expectation. The user may want one segment covering exactly, for example:

a) the whole complex of buildings;

b) each separate module;

c) each roof;

d) each side of the roof.

The characteristics of the segmentation are highly dependent on application (e.g., the user may want to quantify the area's commitment due to an environmental disaster instead of precise identification of urban objects). So the complexity increases to the development of an unique algorithm to solve all cases.

Once a method is defined and the user has the correct parameters of segmentation, there are factors that still deteriorates the results. Very high resolution images are easily available in volume, and the automation goes from a comfort to a necessity. Therefore the closest results has been sought to provide less post-processing. Some

| (a) | (b) | (c) |
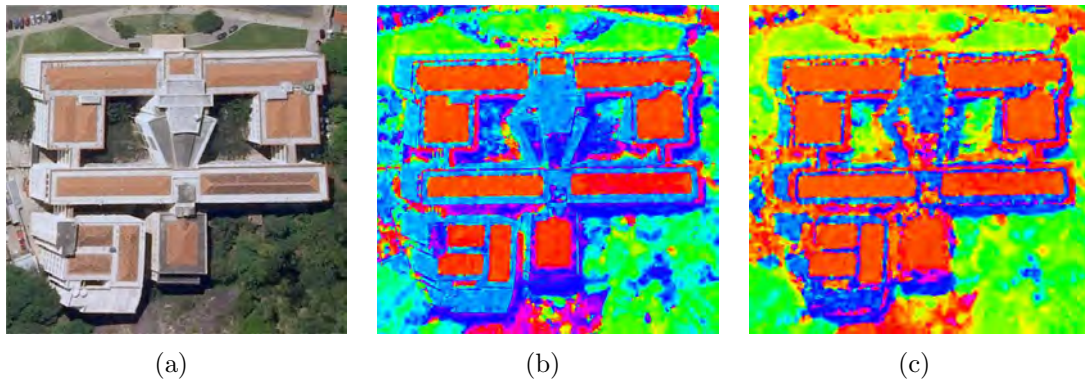
Figure 2.1 - (a) Original image, (b) hue image obtained with SegmenTHIS and (c) normalized hue channel.

problems like image geometric rectification and defects depends on extra information or advanced learning.

A segmentation result can be considered satisfactory when it implies in a reduced need for post-processing activities. Some problems, such as defects and geometric rectification of images, depends on extra information or advanced learning.

Using the original image of Figure 2.1(a), the hue image in Figure 2.1(b) is a clear product to human identification, but the automated process is quite difficult. The hue is being extracted directly from the original image. The building is geometrically well-behaved, having lots of right angles and symmetry. The hue is the source of information of the segmentation algorithm, so it is not possible to get more than exists on the working channel (i.e., hue). If the distinct parts can be clearly identified, the segmentation can succeed.

(a) Segmentation in gray   (b) Segmentation colored with hue mean
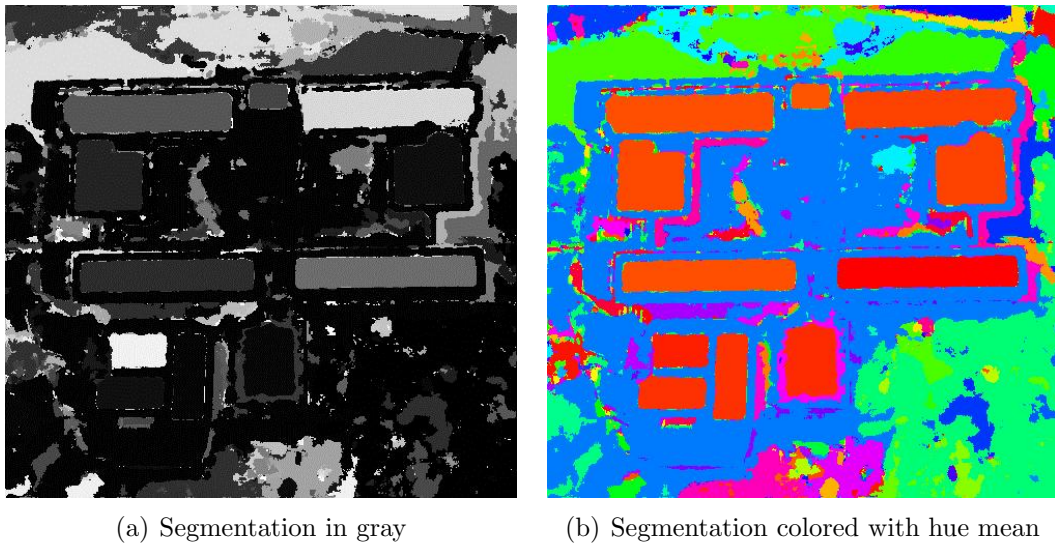
Figure 2.2 - Segmentation using threshold=50.

The question about what information is available always exist, so this example is showing a result with the hue segmenter developed for this research and called SegmenTHIS. In Figure 2.2 two images are representing the same segmentation result, but Figure 2.2(a) is the output in gray and Figure 2.2(b) is a colored version, where the color is the mean of the hue of each segment. If the segment contains a variation of colors (this can happen when using higher value of threshold or when there are shaded colors and the algorithm chosen a variable mean). The gray colored classification output is a file with IDs and, in fact, it was plotted with adjusted contrast.

Presenting a hue segmentation by its mean value of hue can lead to a visual representation close to the original hue image when using a low segmentation threshold value. A low threshold value implies in a low tolerance variation and produces a segmentation with a greater number of segments than when using larger values.

18

Figure 2.1(c) is the hue image after a normalization with mean $\overline{\phi} = 127$ and the standard deviation $s = 45$. This result depend on what is to be considered the foreground and what is the background. An image is acceptable if the objective is to identify the contrasting objects (in any sense, e.g., band value, texture) present in it.

(a) Segmentation in gray   (b) Segmentation in colors
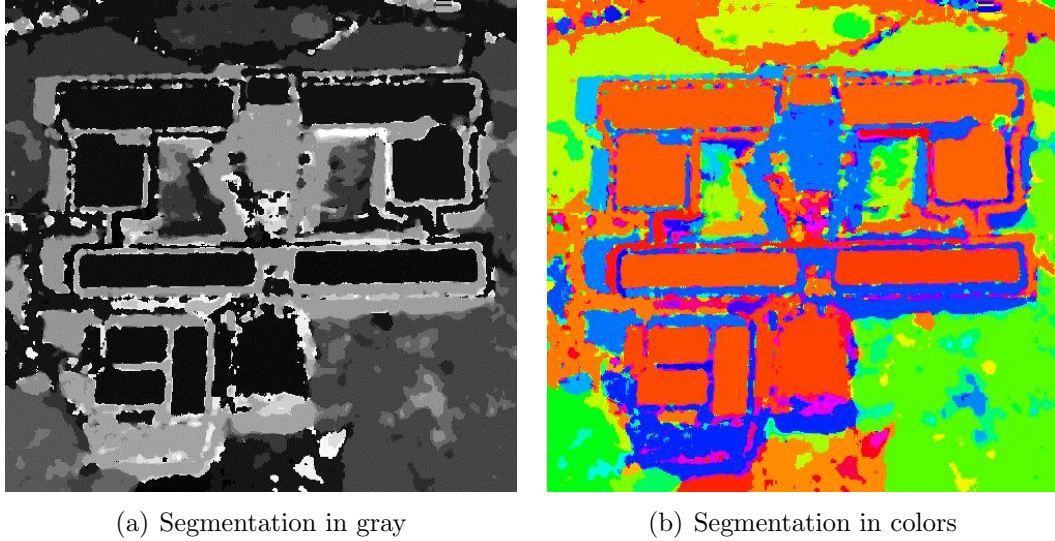
Figure 2.3 - Segmentation of normalized image using threshold=30.

The resulting segmentation for the normalized hue image is shown in Figure 2.3, where the segmentation grouped the pixels based on the thresholding of 30.

2.3 From Segments to Objects

Homogeneity in spatial and spectral characteristics of the objects results in better segmentation. Higher-resolution images, however, provides heterogeneous spectral and spatial behaviors for the same object, and this complicates the segmentation of objects (UBA, 2009).

For example, a house in a LANDSAT/TM sensor was represented by a pixel with sides of 30 meters. Almost any medium house was represented by only one pixel or, by mixture, in four pixels. The shape of a pixel is simple and all houses, undependable on the actual physical shape, was represented by the same square. Satellites with pixel size about 50 cm produces much more complex shapes, because very high resolution sensors captures real object details with high accuracy. The same happens with spectral information: there is much more information about the single object,

as the spatial resolution is increased; and the sensors are becoming much more sophisticated and advanced, being available images with higher radiometric and spectral resolutions. Thus, high resolution images makes the segmentation harder. The accuracy on segmentation affects the quality and performance of object-based image analysis methods (IKONOMATAKIS et al., 1997).

2.3.1 Techniques used to convert segments to objects

Objects are the goal of an object-based approach and so we use segmentation as a step to get there. The production of objects may be accomplished promptly by the use of a segmentation method. If this is the case, we do not need to go further anymore: segments already are objects. But if the result is not satisfactory, then a new procedure must give place. Sometimes the search for a solution is not possible by changing the parameters that are available on the segmentation algorithm. The difference between segments and objects is the meaning of the grouping: if grouped as a meaningful area on the real world – even not being the whole real world object – the segment can be considered an object. So, to achieve this level of product, producing a useful segmentation, an analysis needs to be done (and this justifies the reason for the inclusion of the module of shape analysis in SegmenTHIS).

Segmentation is a complex step in digital image processing, and result in a product that makes it possible to extract features, based on any information the images gives, in order to characterize each object, allowing further classification. These attributes are dependent on the segmentation produced, and sometimes these results are not adequate.

2.3.2 Resegmentation

As we have seen, the high spectral and spatial heterogeneity of high resolution inputs makes it more difficult to segment the image, and some techniques have been developed to get a better result, e.g. multi-resolution segmentation. To reorganize a set of segments created by a over-segmented procedure, Korting (2008) used a resegmentation procedure.

Resegmentation is the technique of producing a new segmentation based on a previous one. This way, instead as having pixels as input and segments as output, we can define a resegmentation process as a new segmentation, that has segments as input, including or not the pixels information, and having a new set of segments as output (KORTING et al., 2011; KORTING et al., 2010). Defining this way, and according to

Zhang (2009), resegmentation is a method with a high cognitive level.

In segmentation, pixels are used to produce segments, but in the approach being proposed in this thesis we deal first with pixels, but later with segments, making a kind of resegmentation, but this is made internally.

2.3.3 Shape refinement

The segments may not match objects due to the inherent noise on the image. However when segments are close to the expected but with some noise, a shape refinement may be required so as to allow the shapes to be improved. When we visually identify image segments, we could make shape inferences and force a matching with a pre-known shape.

Both shape refinement and image resegmentation are techniques that search for the modification of previously processed segments. If the segments are not considered objects (see Section 2.3.2), then refinement is not what should be done, but resegmentation becomes more appropriate. In summary, the main difference between these two approaches is the meaning of what a segment is.

2.4 From Objects to Image Understanding

According to Les (2001), understanding is the assignment of meaning to a data by the interpretation, and shape understanding and image understanding are similar (but not equal) fields. Image understanding of a segmented image is concerned with interpretation of scenes of the real world, which have a strong relation with spatial orientation of the shapes and considering the context.

Automation, in imagery applications, requires the system to get information from the available data. Scene understanding achieves the highest cognitive level in the field known (or being presented) as "image engineering"[1] process.

Segmentation process deals with a large amount of pixels and results in groupings associated with some attributes. Though there are some variations to the output of the segmentation, the output can be a new image with its pixels grouped by segments and named by unique identifiers.

The segmentation process is one of the tasks that make up image processing and

---

[1]The term "image engineering" is presented by Zhang (2009) to refer to a new discipline covering all subjects related to images.

create sets with the aim of grouping homogeneous regions that comprise and compose an object. It is framed as a mid-level processing because it inputs images but outputs attributes extracted from them.

The primitive operations of low-level processes performed on images by coding, projection, transforms, restorations and enhancements gives place to higher cognitive level operation of segmentation, representation, description, texture and shape analysis (ZHANG, 2009) The highest level, according to this same author, is the so-called image understanding. Image understanding, the higher abstraction level, includes image fusion, stereo vision, three-dimensional shape information recovery, matching, content-based image retrieval and understanding. Image techniques grouped on inter-connected layers show that the more use of concepts of objects instead of pixels, the more abstraction and smaller data volume to deal with.

Several approaches have been suggested in order to recognize the objects of interest located in the scene. Shape can be distinguished from its surroundings by its outline, but there are other ways to recognize objects as by its texture, color and context information. Some objects may need contour with internal texture information, but many others could be recognized only by their contour shapes.

Snakes is a simple deformable model of active contour that can modify closed polygonal lines from a starting shape in order to match the edge of a region. This model makes the shape to keep changing from the initial configuration to the edge refinement iteratively and driven by an image field.

Using this mechanism of refinement in the process of segmentation we can expect to have more precision. But if the contour could be modeled by a set of shape rules and, at the same time, could satisfy the segmentation main rules it is possible to achieve better segmentation results in terms of expected shapes, if a classification of the shape could be, somehow, defined internally, during the image segmentation process.

After a trial of some techniques the turning function – and other derived representations, such as tangent angle, cumulative angle and periodic cumulative angle (VELTKAMP; HAGEDOORN, 2001), –showed some interesting properties. The study of this shape representation revealed a great potential to make rules, although some caution must be taken. It were proven to be feasible by a number of required properties we can derive. Turning functions will be explained and explored on section 3.6.

A template-based recognition technique search for similarities between two targets of the same kind. The use of turning function in such approach depends on a database with several cases for each object and there is a matching, in terms of shape, when the dissimilarity metric is minimum (and the metric value is below a threshold).

A scene comprehension system has a set of fundamental operations, including segmentation, considered one of the most important operation for the information extraction (UBA, 2009).

3 METHODOLOGY

3.1 Introduction

Many image concepts was presented in Chapter 2 for the development of the methodology to be presented in the following sections. The framework of the methodology of the present thesis is pictured in Figure 3.1, with the body of the methodology in the central column (the light-blue and yellow boxes) and the green boxes representing the partial conclusion, used as input to the development and implementation of the SegmenTHIS prototype with shape analysis (the yellow box, described in Section 3.8). Each box in this diagram is being described in the sections of this chapter.
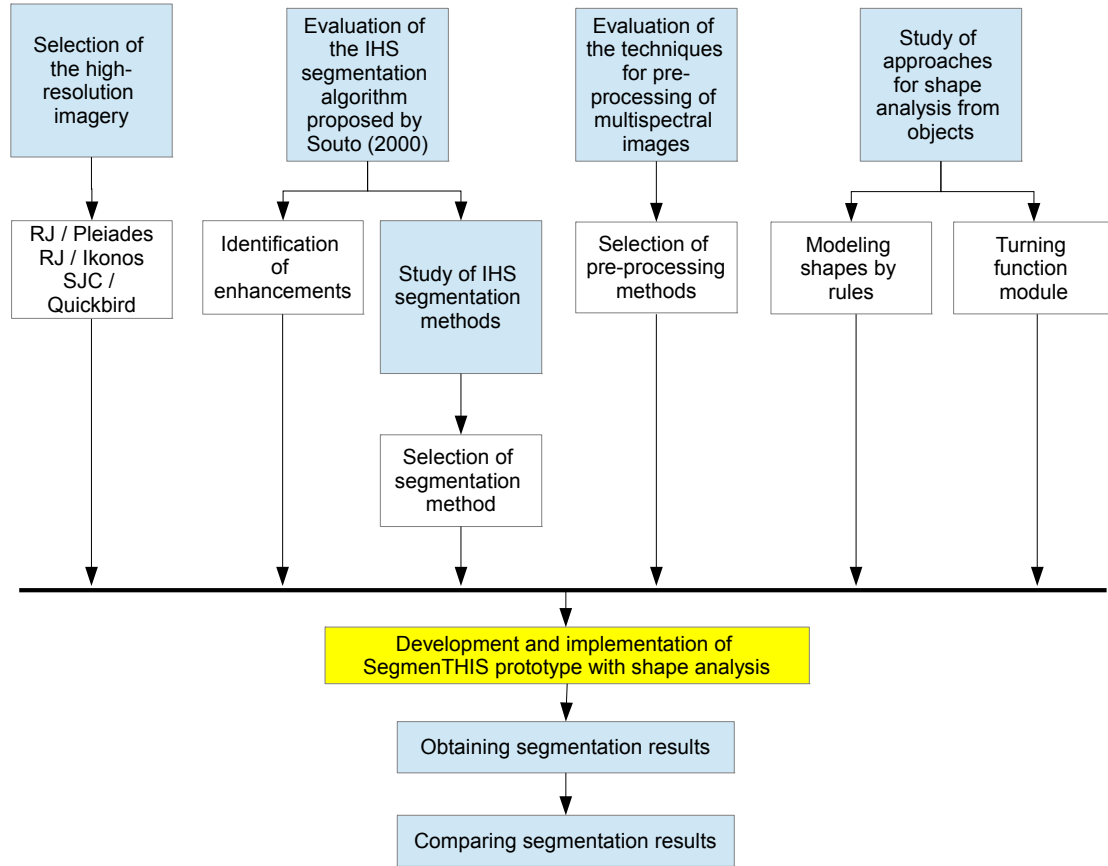
Figure 3.1 - Methodology overview diagram.

The first was the selection of the high-resolution imagery. This selection defines the validity of the proposed method, since images where the objects are not present or do not have useful information on the highlighted hue channel will not produce

25

useful results for methods such as this, based on hue. The images selected are inputs to SegmenTHIS prototype.

The evaluation of the IHS color space segmentation algorithm proposed by Souto (2000) was the starting point, where the advantages and disadvantages was considered in order to identify possible enhancements to the existing prototype in IDL language. These needs was used as information in the development stage.

The evaluation of the techniques for pre-processing of multispectral images was studied to allow the selection of methods to enhance and make feasible the use of the selected base algorithm.

The study of IHS segmentation methods is a step of the research responsible to search for alternatives in using hue, saturation and intensity channels from multispectral bands, and the effects caused by pre-processing suggested in the previous stage. This resulted in the use of Moik method as the base method and in the development of two new methods: Sin-Cos method and S-Weighted method.

An intensive work was given in the study of the approaches for shape analysis of the objects. This stage is the responsible for the definition of a shape analysis module and the proposal of modeling rules for the shapes. The inclusion of the shape analysis module and how to deal with shapes inside the segmenter is the key idea of the thesis. The shape module is not bound and does not depends on hue segmentation, but seems to give very good results in any kind of segmentation or object detection, however, it was not tested and is only a suggestion for future work. Results and comparisons are presented in the Chapter 4.

3.2 Selection of the high-resolution imagery

The images used in this work was selected from: IKONOS, Plèiades and Quickbird satellite images. The reason for choosing these images was the availability of the images, the resolution characteristics (shown in Tables 3.1,3.2,3.3) and the peculiarities of each one for being appropriate for urban areas study. The geographic areas includes Rio de Janeiro (RJ) and São José dos Campos (SP) in Brazil, because of the acquaintance with the region, needed to identify visually targets and make manual identification of objects.

Plèiades-1A satellite sensor characteristics are presented in Table 3.1 (SATELLITE IMAGING CORPORATION, 2013), IKONOS sensor characteristics in Table 3.2 (COOK et al., 2001) and Quickbird sensor characteristics in Table 3.3. In the table, the accu-

Table 3.1 - Plèiades-1 satellite sensor characteristics

Imagery Products	50 cm	black and white
	50 cm	color
	200 cm	multispectral
	Bundle:	50 cm B&W and
		200 cm multispectral
Spectral Bands	Pan:	480-830 nm
	Blue:	430-550 nm
	Green:	490-610 nm
	Red:	600-720 nm
	Near Infrared:	750-950 nm
Image Location Accuracy	With GCP:	1m
	Without GCP:	3m
Imaging Swath	20 km at nadir	
Revisit Interval	Daily	
Dynamic Range	12 bit per pixel	

Table 3.2 - IKONOS satellite sensor characteristics

Imagery Products	82 cm	panchromatic
	320 cm	multispectral
26° Off-Nadir	100 cm	panchromatic
	400 cm	multispectral
Spectral Bands	Pan:	525-930 nm
	Blue:	445-515 nm
	Green:	505-595 nm
	Red:	630-700 nm
	Near Infrared:	760-855 nm
Image Location Accuracy	With GCP:	1m
	Without GCP:	3m
Imaging Swath	11.3 km at nadir	
Revisit Interval	3 days	
Dynamic Range	11 bit per pixel	

Table 3.3 - Quickbird satellite sensor characteristics

Imagery Products	61 cm	panchromatic
	244 cm	multispectral
25° Off-Nadir	72 cm	panchromatic
	288 cm	multispectral
Spectral Bands	Pan:	450-900 nm
	Blue:	450-520 nm
	Green:	520-600 nm
	Red:	630-690 nm
	Near Infrared:	760-900 nm
Metric Accuracy	23 m horizontal	
Imaging Swath	16.5 km at nadir	
Revisit Interval	1-3.5 days	
Dynamic Range	11 bit per pixel	

racy without ground control point (GCP) is 3 meters. Four orthorectified spectral data was provided at 0.5-meter resolution and used (in Pleiades examples).

The perceptual transformation shown to be very sensitive to compression degradation, so the uncompressed images are best suited for this purpose.

Some images provided was pre-processed by a pan sharpened image fusion (4-Band Pan-Sharpened) which produced a non-usable hue channel. All images are very clear and with visible excellent quality, but our tests showed the pan-sharpened to be inadequate for our purpose. The RGB-IHS conversion process presented in this thesis exposed a nonlinear transform in the hue channel and greatly impaired the IHS transformed channels, resulting in a channel of H cloudy appearance, despite the high quality of RGB image. So, a more detailed study of the applicability of the images will need new tests and the use of rawer bands.

3.3 Evaluation of the IHS segmentation algorithm proposed by Souto

The IHS segmentation algorithm proposed by Souto (2000) was chosen because it was available, and implemented in IDL, facilitating the new additions that would be required by the object-based proposal. The randomness of the algorithm seems to create different segmentations each time, but this issue depends on the combination of threshold value and scene. The most noticeable limitation was the inability to deal with different parameters. There was only a few adjustable alternatives and so, if there was a bad segmentation result, there was not much to be done.

Some enhancements was selected, as to change default values into parameters, e.g., creating new parameters, to find new ways to change hue data when there is concentration in a region of the (circular) histogram and modifications in the interface.

3.4 Evaluation of the techniques for pre-processing of multispectral images

This section presents the evaluation of techniques for pre-processing. There is a great number of techniques for pre-processing multispectral images, but the use of the IHS color space has its own characteristics and constraints. To synthesize n bands in exactly three IHS channels the method proposed by Moik (1980) is being used. A PCA pre-processing was used for testing and can be useful when there is not many dark areas. These two methods are being presented here as options used to integrate and enhance the base region-growing algorithm.

3.4.1 Multispectral IHS Moik Transform

It is possible to get information from n multispectral bands and synthesize in three channels of IHS space. This was proposed by Moik (1980) by the redistribution of the 2π interval of hue in n parts, instead of the original 3 parts of the RGB space. The equations (3.2), (3.3) and (3.4) present the use of the spectral bands to produce a triple of I, H, S:

$$\bar{\phi}_{1=0}, \quad and \quad \bar{\phi}_k = \bar{\phi}_{i-1} + \frac{2\pi}{n} \tag{3.1}$$

$$I(x,y) = \frac{\max f_k(x,y)}{f_{max}} \tag{3.2}$$

$$H(x,y) = \phi(x,y) = \arctan \frac{\sum_{k=1}^{n} f_k(x,y) \cdot \sin \bar{\phi}_k}{\sum_{k=1}^{n} f_k(x,y) \cdot \cos \bar{\phi}_k} \tag{3.3}$$

$$S(x,y) = 1 - \frac{\min f_k(x,y)}{\max f_k(x,y)} \tag{3.4}$$

for (x,y) the pixel coordinates, n the number of spectral bands, f_{max} is the maximum intensity of all n bands, $\max f_k(x,y)$ is the highest value at the position (x,y) and considering the n bands, $\bar{\phi}_k$ represent a constant value (angle) for each of the kth band.

These equations are implemented on the algorithm used as the basis of this thesis, and therefore are integrated in SegmenTHIS as a method to obtaining hue from

images.

Considering this approach let's suppose the case of four bands. For each pixel, the first band will contribute with 0 in the numeration of the summation and with $f_1(x, y)$ on the denominator. The second band will add $f_2(x, y)$ on numerator and 0 in the denominator, and so on. For the case of 4 bands, the result will be $\arctan((f_2 - f_4)/(f_1 - f_3))$. The results of equation in the segmentation by hue was studied by Souto (2000).

3.4.2   Principal component analysis and normalization

Instead of using the original bands an option is to obtain new channels by the principal component analysis transform.

The information of the original bands allows to obtain new bands by rotations in the feature space[1] and new channels substitute all original information but with the new bands concentrating the maximum possible variance in the first component, and consecutively until the last band has the smallest variance. If the bands are strongly correlated, the last channel is not usable, with variances close to zero. This technique allows the reduction of dimensionality on multispectral imagery because in the first bands you get the principal information of the whole set. The new channels are linearly less correlated, and the first channel is called the "first principal component". Every antecedent channel have a larger variance than its subsequently numbered channel.

The conversion RGB–IHS should provide a result that does not meet the main advantages of choosing the method of hue segmentation: the possibility of removing the intensity information without modifying the other channels, but the use of principal components instead of the bands aims to reduce the incidence of low saturation areas.

Normalization is a new feature available and can be applied when opening the original image or after a processing (e.g., to intensity, saturation or principal component images). To facilitate operation a combination was fixed in 3 basic conditions:

- do nothing on image;
- stretch to $0 \cdots 255$; and

---

[1] According to Jensen (2004), features in the pattern recognition literature are the individual bands of remotely sensed data.

- normalize image to mean $\bar{f} = 127$ and $\sigma_s = 45$.

Example 3.2(b) will be used to analyze the input algorithm. First we need to define the extension and find the local file. After the file selection we need to define what type of principal component normalization we want.

(a) Widget to open image and extracting phase band

(b) Original picture
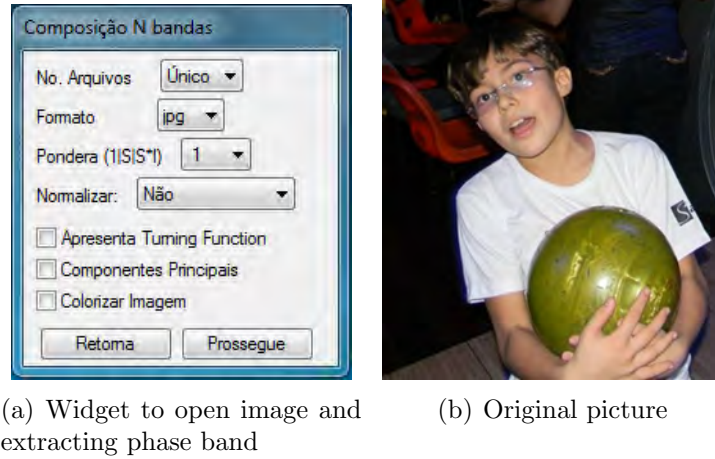
Figure 3.2 - Opening an image in SegmenTHIS.

After checking the "principal component" option on 3.2(a), nine variants of the process become available by the program, giving to the user the ability to combine three types of original image normalization with three types of principal component normalization.

3.4.3 Choices for normalization

The utilization of principal component to extract the hue needed a prior standardization. Figures 3.3 are very poor and we can visually detect this fact. This happens because the principal component process leads to new distributions of the values unevenly distributed and the normalization will change digital numbers to a new mean and variance level. This minimizes the chance of the hue band to be focused around a limited zone of the color cone and reallocates the data.

Figure 3.3 shows three types of normalization available: 3.3(a), 3.3(b) and 3.3(c), but using the original image without histogram normalization.
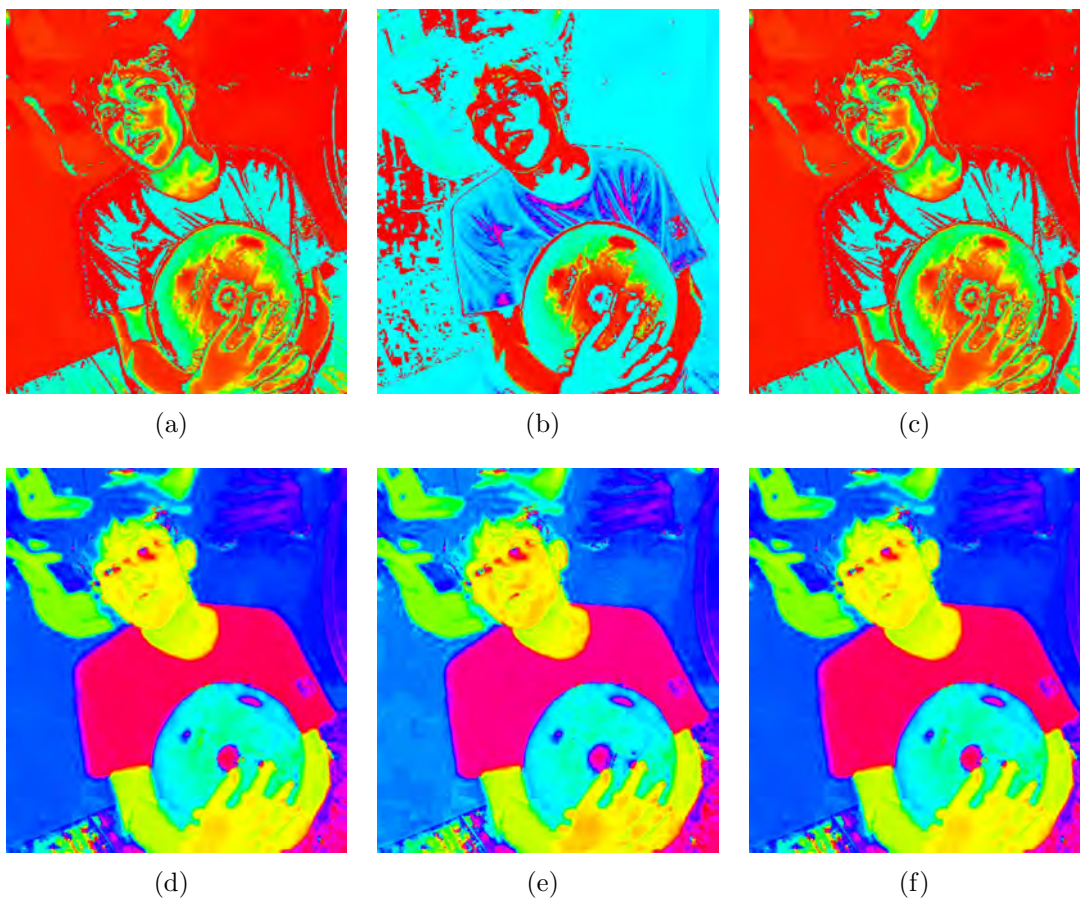
Figure 3.3 - Hue from HSV transformation of RGB composition of three first principal components – with PC bands not changed: (a) no normalization, (b) original image normalized, (c) linear stretch is to be applied to original image; and with normalized PC bands (d) no normalization, (e) original image normalized, (f) linear stretch is to be applied to original image.

The choices available for each image are:

a) no change;

b) a normalization for mean of 45 and standard deviation of 127 for a byte image; and

c) histogram stretching.

In all cases the normalization changes depends on the original image initial contrast. If the original image is already stretched there will be no differences between histogram stretching and no change cases, and there will be a reduction to six distinct combinations.

This does not occur with principal components. It would be unlikely that the bands of principal component have the same contrast and brightness because the first component achieves the maximum spread in data then the second component achieves the second largest scattering and so on.

(a)   (b)   (c)
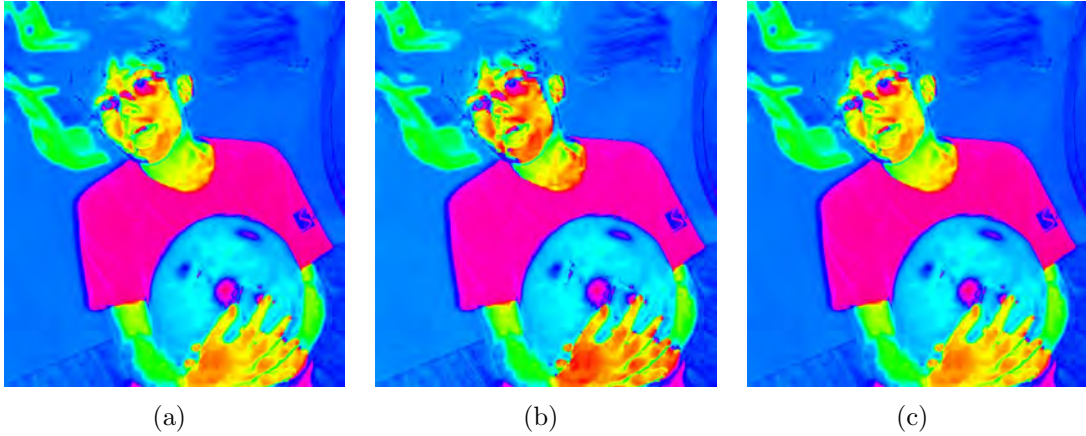
Figure 3.4 - Hue from HSV transformation of RGB composition of three first principal components: (a) no normalization, (b) original image normalized, (c) linear stretch is to be applied to original image; and with stretched PC bands.

3.5 Development of IHS synthetic images

In the IHS color space, the concentration of intensity on the sensed signal in one channel (i.e., intensity) attenuate the effects of glare, gradients and shadows in the other two channels, hue and saturation. In IHS, the hue is being used to identify

objects that are distinguishable by this attribute, and the same principle of detection is being extended to multi-band images.

The use of hue is not suitable for all cases because it depends on the difference on hue of each object and its surrounding. As an example, if there is contrast in hue channel between a given object and the background, but it is touching other similar objects, the individual shape detection may fail as the shape will enclose all the overlapping objects. Another case of failure is the existence of regions of gray color (either in background or foreground).

We propose two alternative heuristics:

    a) a sine and cosine method for the use of any regular segmentation algorithm to deal with hue; and

    b) a weighted method to obtain a synthetic hue by weighting hue by saturation, for using more than 3 band.

The sine and cosine method splits the hue channel into new channels and these channels can be used as regular intensity image because there is no discontinuities for close angles. This heuristic is based on the idea that when one channel performs poorly the other is being very thorough.

The saturation-weighted method is only useful if the image has more than 3 spectral bands, because the chance to have simultaneously all the bands with the same (or even very close) values becomes more difficult to happen. The saturation-weighted hue (equation 3.9) calculates a new hue as a vector sum of hues of all bands, favoring the hue values corresponding to the highest values of saturation. Considering b multispectral bands combined in n IHS compositions, then the number of combinations $n = C_b^n$ is the number of hues (and saturation and intensity bands) to be used.

To improve the process it is being presented in the next sections a way to produce sin-cos and saturation-weighted synthetic hue channels for multispectral imagery.

3.5.1 Sin-Cos method

A heuristic was used to perform the segmentation based on hue information but using any available segmenter. This alternative approach was developed to get the hue produced and create the new components: sine ($f_{sin}(x,y) = \sin(f_\phi(x,y))$) and cosine ($f_{cos} = \cos(f_\phi(x,y))$) channels, where f_ϕ is the hue image. Using these two images

g_{sin} and g_{cos} with any classical segmenter produces two segmented images: g_{sin} and g_{cos}. A merge algorithm is used to join these two results in a single segmentation g_ϕ. This approach is being presented in Moik, but it depends on the existence of a hue channel, regardless of the method that has generated it (so it could be included in the next method too).
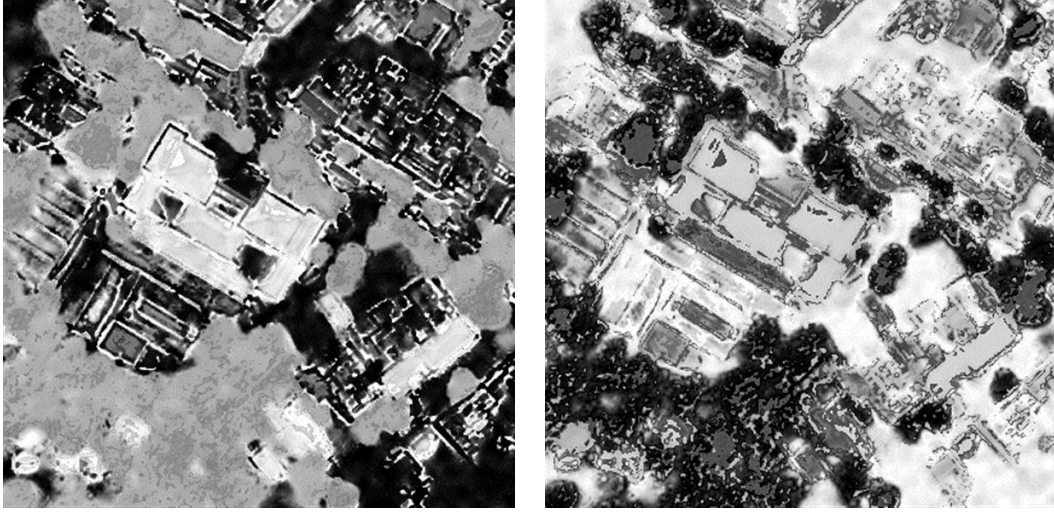
(a) Sine image of synthetic hue   (b) Cosine image of synthetic hue

Figure 3.5 - Sine and cosine images, built from the synthetic hue image.

Both, sine and cosine are periodic functions, with period equal to 2π. Sine and cosine are continuous and differentiable functions in $f_\phi \in \Re$. So, if f_ϕ is a differentiable function then $\sin(f_\phi)$ is differentiable and also is $\cos(f_\phi)$. This way, creating these two channels with the sine and cosine functions, the discontinuous function f_ϕ is transformed into two continuous ones (FENG et al., 2001).

The existence of both sine and cosine values are sufficient to get back the hue channel. The theoretic idea being used is that we can use both sine and cosine images to get the information of ϕ, and the smoothness on ϕ implies a smoothness in both $\sin(\phi)$ and $\cos(\phi)$.

If we call $g_\phi(f_\phi)$ the segmentation g_ϕ of the hue image f_ϕ, and if $g_{sin}(f_{\sin\phi})$ is the segmentation of the sine image of the hue and if $g_{cos}(f_{\cos\phi})$ is the segmentation of the cosine of the hue image, then we propose, as an heuristics, the creation of a new image segmentation $g'_\phi(f_\phi)$.

$$g'_\phi(f_\phi) = g_{sin}(f_{\sin\phi}) \bigcap g_{cos}(f_{\cos\phi}) \tag{3.5}$$

$$g'_\phi(f_\phi) \sim g_\phi(f_\phi) \tag{3.6}$$

Equation (3.5) is the way to get the new Sin-Cos segmentation $g'_\phi(f_\phi)$ from the sine and cosine segmentations by intersection on results. Equation (3.6) is presenting the Sin-Cos Method ($g'_\phi(f_\phi)$) is similar to the Hue-Method $g_\phi(f_\phi)$. The sine and the cosine channels are shown in Figure 3.5.
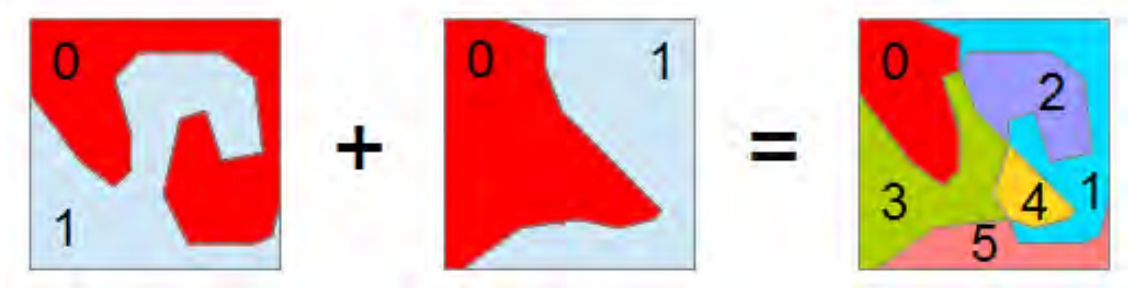
Figure 3.6 - Detail on junction operation for segmentations.

To gather the information of two images in a single image the intersection of Equation (3.5) is shown Figure 3.6. Figure 4.23 represent the intersection operation of segments in both Figure 4.24 and Figure 4.23(a). Two segments on the first picture are joined to the two segments on the second image and the result is a final image with six segments. The junction of segmentations, as those exemplified in Figures 3.7 for the composition of the new segmentation from sine and cosine images, can be used to gather information from multiple files into a single result.
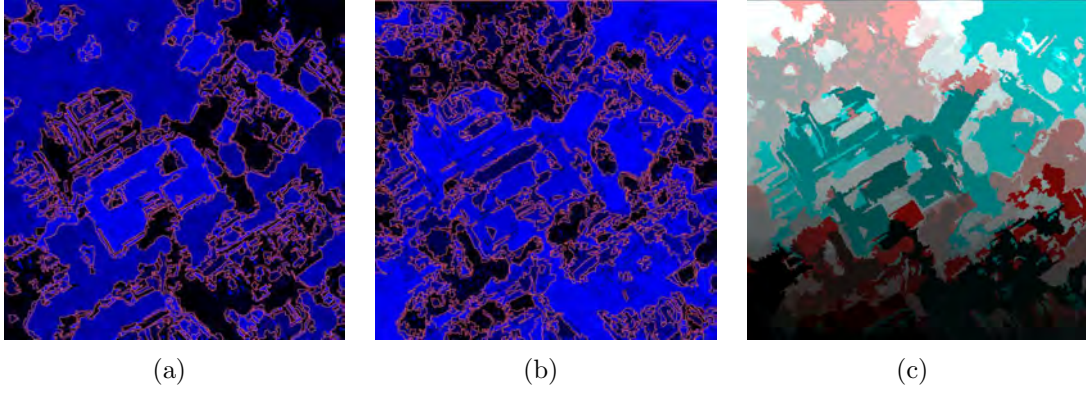
Figure 3.7 - Sin-Cos segmentation method in Spring: (a) segmentation of sine; (b) segmentation of cosine; and (c) result of Sin-Cos.

3.5.2 S-Weighted segmentation method

The use of three spectral bands as input of the RGB-IHS transform lead to three channels IHS where low saturation implies in imprecision in hue information of the hue channel. Null saturation happens whenever the three band intensities have the same value. Hue information on areas with low or null saturation, as in very contrasted black and white effect of white ship and dark shadow, can lead to unpredictable results, because hue in these kind of areas is very unstable and even indefinite. To minimize this problem we propose a new strategy with the use of multiple bands and define a method to integrate multiple IHS from combinations, radiometry manipulation on these bands, execution of multiple segmentations and finally making a weighted junction of the multiple results.

The combination of m multispectral images produces $n = C_m^3$ colored compositions and we can obtain n hue-saturation-intensity images. Souto (2000) made his studies based on the multispectral determination of a single set of hue-saturation-intensity images by using the equation proposed by (MOIK, 1980). We are suggesting an alternative way to obtain a synthetic hue from all multispectral channels.

Let the hue, saturation and intensity of the point (x, y), on the bth channel, be $H(x, y, b)$, $S(x, y, b)$ and $I(x, y, b)$, respectively. The hue $H(x, y, b)$ is the complex argument (phase) of the complex number z, obtained by equation (3.7), where $\rho = |z|$ is the complex modulus, i.e., the saturation[2].

---

[2]Considering the geometric representations of IHS color space, the complex notation can be used to represent hue (angle) and saturation (the modulus).

$$z = \rho\left(\cos(H(x,y,b)) + j \cdot \sin(H(x,y,b))\right) = \rho \cdot e^{j \cdot H(x,y,b)} \tag{3.7}$$

Then, if we do a weighted average by the saturation, we'll obtain equation (3.8). Unfortunately, not all problems end, since the result can be an undefined value. All values of saturation on the n bands being zero or the numerator being zero leads to indefiniteness, however this is intrinsic to IHS system, so we tested the new approach.

$$z(x,y) = \frac{\displaystyle\sum_{b=1}^{n} S(x,y,b) \cdot (\cos(H(x,y,b)) + j \cdot \sin(H(x,y,b)))}{\displaystyle\sum_{b=1}^{n} S(x,y,b)} \tag{3.8}$$

From Equation (3.8) we can obtain Equation (3.9), a new synthetic hue channel, as proposed and implemented in this thesis, based on multiple values of hue and saturation from the same pixels. This method is intended to be used with multispectral imagery of four or more bands. Equation (3.10) adds intensity information to equation (3.9) and is an alternative way to compute hue.

$$H'(x,y) = \arctan\left(\frac{\displaystyle\sum_{b=1}^{n} (S(x,y,b) \cdot \sin(H(x,y,b)))}{\displaystyle\sum_{b=1}^{n} (S(x,y,b) \cdot \cos(H(x,y,b)))}\right) \tag{3.9}$$

$$H'(x,y) = \arctan\left(\frac{\displaystyle\sum_{b=1}^{n} (S(x,y,b) \cdot I(x,y,b) \cdot \sin(H(x,y,b)))}{\displaystyle\sum_{b=1}^{n} (S(x,y,b) \cdot I(x,y,b) \cdot \cos(H(x,y,b)))}\right) \tag{3.10}$$

This method is designed to reduce image regions of high instability, when hue is of little relevance on the definition of the color. The multispectral IHS transform proposed by Moik considers all values as having the same weight, regardless of the values of brightness or saturation are low or high, i.e., even if the value of hue is spurious.

The idea behind the implementation is to give more weight for hues corresponding

to higher saturation (and intensity). This serves to focus on the areas with a better defined hue. The implementation of these equations produce a synthetic image hue channel. By giving more weight to combinations with higher saturation and ignoring those with zero saturation, this equation produces a false hue channel. This phase channel not physically meaningful but tends to preserve or to be closer to the phase values of the highly saturated areas. If the same geographic position have the same hue value in all channels, the value will be that one. If, however, the hue is different, the weighted average will approach the hue value (or average) corresponding to highly saturated pixels.

3.6   Study of approaches for shape analysis from objects

This section presents the way used in this thesis to represent shapes for shape analysis. The process of identifying and manipulating the contours of the segments, that is being proposed, is the use of turning functions by direct modifications using this representation. This new approach is being suggested for being necessary to meet the proposed requirements of achieving an operational object-based segmentation.

3.6.1   Turning Function

Turning function is useful for matching (LEW, 2001). The angles can be represented by differences, by the tangent angle of the curve with a reference or by the cumulative angle (in the latter case, a turning function can be called a cumulative angle function). For polylines, the turning function $\theta_A(s)$ is a piecewise constant function and gives the angle between the counterclockwise tangent and the x-axis as a function of the arc lengths s, increasing in left turns and decreasing in right turns.

A turning function can also be called turning angle function (RANGAYYAN et al., 2008). The continuous version is called ψ-s curve (ZHANG, 2009; BALLARD; BROWN, 1982) and it is not limited by a finite number of vertices as the others. All this family of curves are represented on an arc length versus tangent angle plot, consisting on uni-dimensional descriptions of two-dimensional shapes. The function curve is generated by traversing the boundaries and obtaining the angle between a reference line and the tangent of that point to the boundary.

An image segment is a closed shape whose boundaries can be represented as a polygon. Any finite two-dimensional polygon represented in the Euclidean plane can be converted into a turning function (VOLOTAO et al., 2010b).

3.6.2 Turning angle space
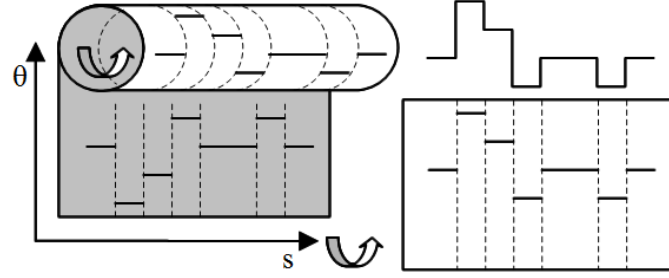
Figure 3.8 - Turning function space for polygons.

The turning function can represent polygons and this thesis focuses on closed polygons, and we use a rectangularly confined space to represent any turning function in the plane. The distances (X direction) and angles (Y direction) are represented in the Cartesian axes, which coincide with the sides of the rectangle. Figure 3.8 shows the representation of any turning angle space as a cylindrical space. This is due to the cyclic nature of the turning angle component, where $2\pi \equiv 0$. To represent this space we can think on a bending of the turning function planar space.
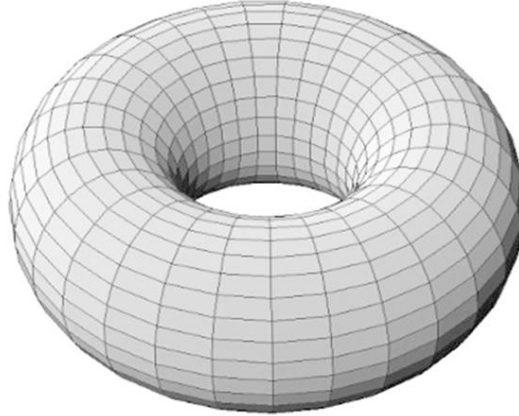
Figure 3.9 - The turning function of any simple 2D shape might be represented by the space defined by the surface of a ring torus. Any polygon can be represented as a turning function. with the starting point connected to the end point, and the maximum value of the function is adjacent to the minimum value. The knowledge of the geometry of this area is important for the characterization of shapes.

For shapes, the cumulative turning angle space is a ring-torus shaped space and the turning function takes 3.10. To each position in the toroidal axis corresponds one and only one value in poloidal axis. The size depends on the polygon length or can be normalized. The normalized polygon makes it independent of scale.

As the polygon rotates, its turning function on the ring-torus space slides over the small circle (the transversal section) with the same angle variation all over the function. A small circle in the surface of revolution revolves about a coplanar and non-intersecting axis of rotation. The roll around this small circle is named as poloidal slide, for clarity. The poloidal slide represents the rotation of the shape. The distance from the center of the small circle and the axis of rotation defines the big circle and the roll of the turning function by an angle over the big circle and this shift is named as toroidal slide. The toroidal slide defines the shift on the starting point of the shape.
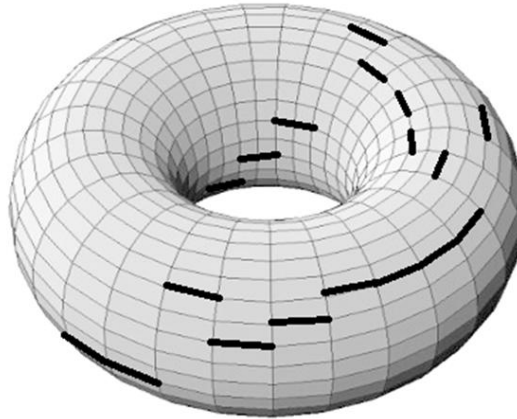
Figure 3.10 - A length normalized turning function is being represented by black lines on the external surface of the torus-ring space in this 3D view.

Shape has three invariants: rotation, translation and scale. So the origin of a shape can be any point. If the perimeter is normalized for all shapes it becomes independent on the size. This is a necessary operation for shape comparison. In the present method there is no need to normalize the perimeter for the torus depends on the contour length and the length of the toroidal circle is exactly the polygon perimeter. However, if normalized all shapes can be compared (Figure 3.11 and Figure 3.12).

Any polygon can be represented as a turning function with the starting point connected to the end point, and the maximum value of the function is adjacent to the

minimum value. The knowledge of the geometry of this area is important for the characterization of shapes (Figure 3.10).
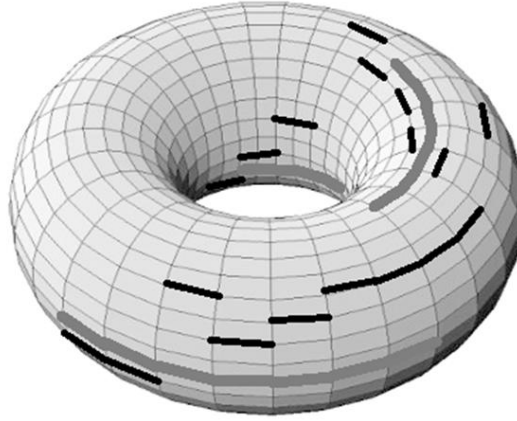
Figure 3.11 - Two turning functions represented on the same ring torus space: one in black and the other in gray.

An example of a polygon as a turning function is presented in Figure 3.10. The normalized polygon makes it independent of scale. As the polygon rotates, its turning function on the ring-torus space slides over the small circle (the transversal section) with the same angle variation all over the function. A small circle in the surface of revolution revolves about a coplanar and non-intersecting axis of rotation. For simplicity, the roll around this small circle is named as poloidal slide. The poloidal slide represent the rotation of the shape. The distance from the center of the small circle and the axis of rotation defines the big circle and the roll of the turning function by an angle over the big circle and this shift is named as toroidal slide. The toroidal slide defines the shift on the starting point of the shape.

Sliding in both directions (poloidal and toroidal) does not change the shape (Figure 3.12). The slide of the gray is represented by the new red turning function. The metric used to measure distance between two curves calculates all distance sums between the curves and finds the minimum between the given functions, one sliding freely over each other.
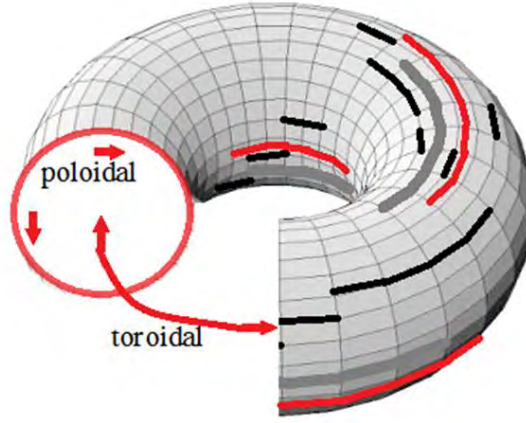
Figure 3.12 - Turning function poloidal and toroidal slides.

3.6.3 Turning function plot

The contours of the 2-dimensional digital objects (i.e., segments) can be converted into a phase representation (i.e., turning function). Considering this fact we are using angles two times on the environment of analysis, i.e., hue and turning functions's θ angles.

This is a new approach and is being presented to allow us to make several operations on segments that can be acquired from an image. Moreover, this new use is a way to allow modifications on contours and to make the corresponding shape to adhere to a given behavior. This model is very simple and intuitive when dealing with closed and convex polygons. A triangle or a rectangle are the first examples we can think about. Swimming pools and house roofs are abundant in the basic rectangular shape. This common example can be used to intuitively demonstrate how this method can work and is consistent.

If we consider a spiral-shaped contour and if we did not normalize the angle to the $[0, 2\pi)$ interval, the angles would have no positive nor negative limit. Length and angle normalized turning functions can provide a shape signature and a limited representation with no lose of accuracy.

We use normalization on both angular and linear components of turning function, considering the circularly aspect of the sequences of values in the intervals $s : [0, 1]$, with $s_0 \equiv s_1$, and $\theta : [0, 2\pi[$. Note that s_0 and s_1 are the edges of the polygon and $\theta(s_0) \equiv \theta(s_1)$ because shapes are closed polygons. Figure 3.13 is a generic

43
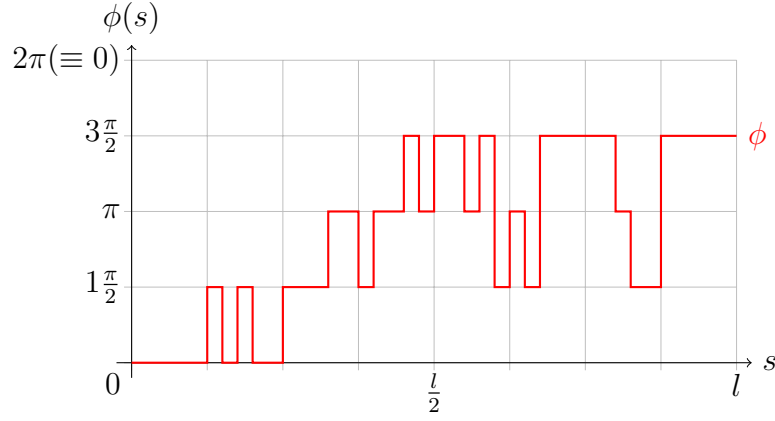
Figure 3.13 - Example of a turning function graphic.

representation of a polygon. In this case of four angles $(0, \pi/2, \pi, 3\pi/2)$ to be a closed polygon the condition (to be presented in detail in 3.6.5) o closure is that all length component of the entire s have equal sum for each of the four angles.

We are defining the generic term turning function as a vector representation of a curve by the use of an angle (absolute or relative to the last) and a length (absolute or relative to the last). With absolute angle we mean an angle relative to a fixed reference (e.g., the horizontal, a given direction or the first side direction) and with relative angle we mean the angle between the tangent on a point of the continuous curve (or each side of the polygon) and a reference: the last one in order or a reference, usually the horizontal, a given direction or the direction of the curve starting point. Absolute length is the total or the cumulative length of the curve or polygon path and relative length is the length of each side of the curve or polygon. Other names for the same (or similar) representation of curves in the scientific literature are (OTTERLOO, 1991; PAVLIDIS, 1978; LONCARIC, 1998; FOSTER et al., 1993; ZHANG; LU, 2004; VOLOTAO et al., 2010a):

a) turning function;

b) turning angle;

c) turning angle function,

d) tangent angle;

e) psi-s curve;

f) angular function; and

44

g) cumulative angular function.

The definition depends if considering the continuous or discrete curve, but for images we always refer to discrete curves, which means dealing only with polygons.

For modeling with turning functions a known reference shape can be found to fit to the model, what is addressed by Liu (2004). Shape regularization or beautification (IGARASHI et al., 1997; LLADÓS et al., 2002) are techniques that attempts to increase:

a) connectedness;

b) perpendicularity;

c) congruence; and

d) symmetry.

3.6.4 Working on the turning function space geomtery

The planar space of the turning function[3] as presented in this chapter, have a notable difference to the Euclidean space, because one dimension does not represent a length but the shape angle. Figure 3.14 shows an example to gives the idea of the relations between turning function and geometric properties.

There are several ways to make transforms, and the correspondence presented in this figure is visually presented just to exemplify the use of the metrics related to this thesis and presented in the thesis and in the papers listed in Section 5.2. First it is of key importance to notice that the turning function addresses a tangent angle for each segment, but the vertices does not have its angle defined. This is why the vertical alignment has multiple representations of the same point. The first point is coincident to the last point if the polygon is closed. Starting in the turning function graphic from top, the stair-shaped turning function indicates a square (i.e., the gray square showed in the upper right position). Every point A, B, C, D is represented by its lines, so, as discussed above, multiple representations of the points are being represented, although the point itself is indefined (i.e., there could be a vertical dotted line from 0 to 2π to represent the loci of a point).

The second graphic adds the points P_1, P_2, P_3, P_4 to the same turning function of the square. Some odd thing occurs when dealing with extension of the existing

---

[3] as planar we mean the space used for the usual plane representation, not the torus-shaped 3-Dimensional space presented in 3.6.2)
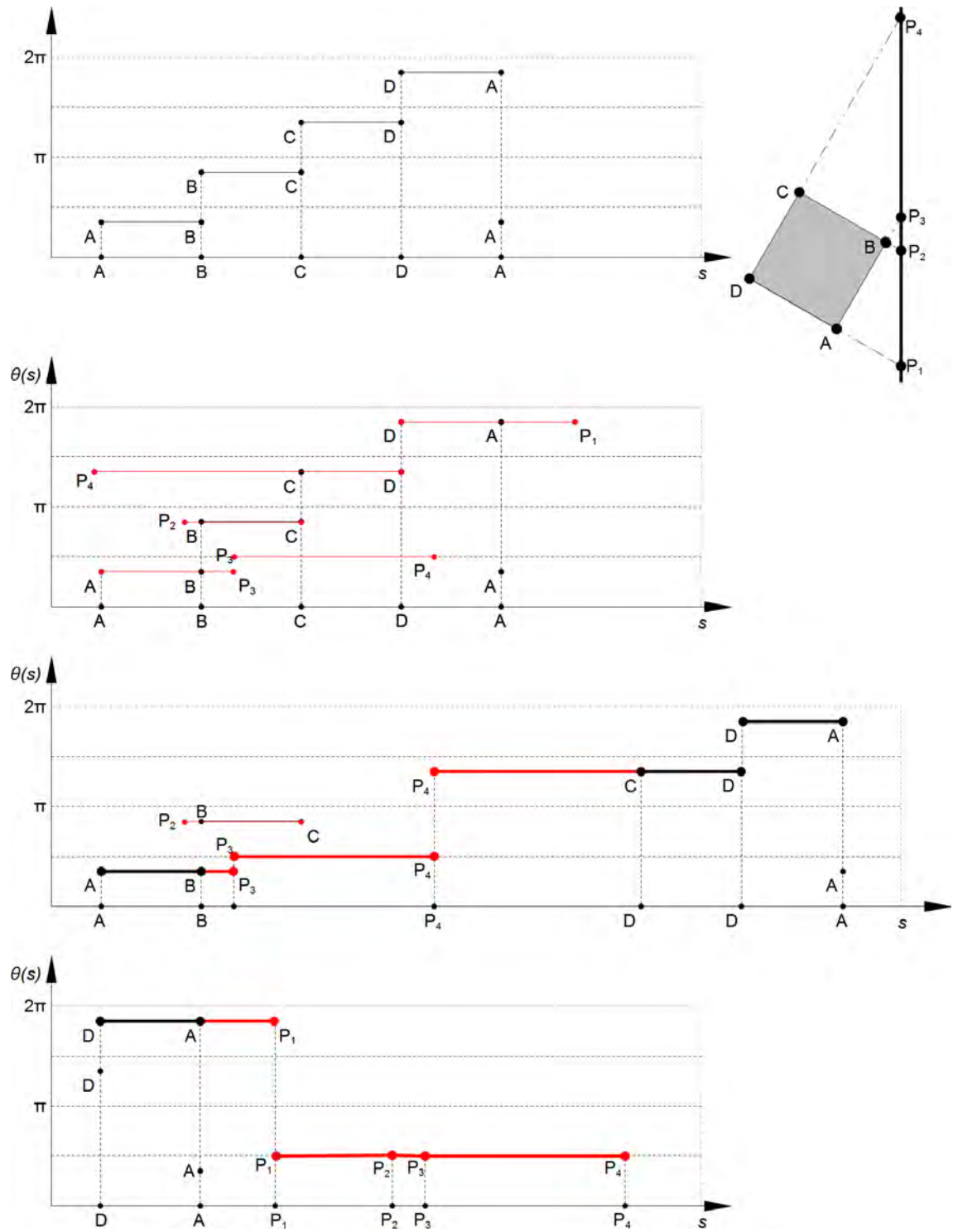
Figure 3.14 - Tangent function representations for a square, a line and projections. This figure illustrates the dealing with tangent function space.

46

lines: Despite the points are well defined by extending by the length (e.g., $\overline{DA}$ and $\overline{AP_1}$ is represented in the turning function graphic by a red dot P_1 and $\overline{DP_1}$ is represented by a red line), it seems that there is no relationship between the loose points P_1, P_2, P_3, P_4.

Trying to explore and to see what happens with P_i points in the third graphic from top in Figure 3.14, the polygon ABP_3P_4CDA is represented by its turning function using real length. Note that the point B is a vertex, so any point crossing the vertical of the B position can be used to represent uniquely another segment. So, the loose segment $\overline{P_2BC}$ is the same segment identified in the reference picture (represented in the upper right part of Figure 3.14). There must be known that the reference is the point B to make the connection between the two polygons. The graphic below represent only two lines $\overline{DP_1}$ and $\overline{P_1P_4}$. The point A is contained in segment $\overline{DP_1}$ and the points P_2 and P_3 are contained in segment $\overline{P_1P_4}$.

We can conclude that the representation is not simple to represent the geometry (e.g., based on the considerations of Figure 3.14) as some issues appears and a geometric consistency is not well established. The development of the graphical properties of turning functions and how to operate them to obtain new information can be developed in future works. It may be possible to find graphical relationship between the shapes, as we discussed roughly because this is outside the scope of the research.

3.6.5 Area of a closed polygon represented by turning function

The turning function can be used to find the inverse transformation and find implicit curves. Turning functions can be represented by a step function $\theta(s)$, where $\theta_i \in [0, 2\pi)$, $u_i(s) = 1$ if $s \in [s_{i-1}, s_i)$ and 0 otherwise, for $i = 1, 2, \cdots, n$, where n is the number of segments and $s_0 \equiv 0$. The step function $\theta(s)$ is the turning function and we formalize it by equation (3.11), where s is the path length and, therefore, we have $s : 0 \to$ perimeter.

$$\theta(s) = \theta_1 u_1(s) + \theta_2 u_2(s) + \cdots + \theta_n u_n(s) \tag{3.11}$$

The area A of a region can be obtained by equation (3.12), where Δs_i is the length of each of the n segments. To use this equation, θ_i is the tangent angle or the cumulative deflection angle of the turning function, not the turning angle.

$$A = \frac{1}{2} \left\{ \sum_{i=1}^{n-2} \left[\Delta s_i \sum_{j=i+1}^{n-1} (\Delta s_j \sin(\theta_j - \theta_i)) \right] \right\} \qquad (3.12)$$

In practical terms the implementation of a turning function filter can be handled by a dichotomy of the parts of the function to recognize whether it is a line segment or not. If the section being analyzed is identified as being a line segment then if it is noisy it must be straightened by the replacement of the resulting vector. Some issues arise in reply to this question related to segment minimum size and tolerance and this something inherent to the definition of the model and will be discussed briefly in the next section.

Accordingly to Les and Les (2008) two-dimensional figures lays on the figure category and consists of polygons, curve polygons and curves. The category of polygons is divided into two categories: the named and the non-named polygons. If we have a 2D figure but not a polygon we have a curve polygon or a curve. So we can say if part of a $\theta(s)$ does not fit a linear segment it better represents a curve.

3.6.6 Identifying shapes

After we extract segments of remote sensing imagery it is then necessary to identify and label the connected components by an unique identification numeric code, i.e., value. Each distinct region become a set of pixels we can call blob. For each blob we can associate a distinct color table, so the blob coloring can represent a randomly defined color or, e.g., the average color, computed by the mean of each band of the segment. The shape of a blob is then extracted and can be represented by its chain code(FREEMAN, 1974).

Consider the following example. Let's use a noiseless representation of a geometrically well defined shape. It is possible to classify the shape by a rule, and in this example we will consider only the angle amount and the symmetry, using turning function values. Consider a shape represented by its turning function with ten elements where the turning angles, numbered from 0 to 9, are $S = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9\}$ and we have $s_0 = s_1 = s_4 = s_5 = -\pi/4$ and all other values of $s_i = +\pi/2$. Analyzing the θ angles set and the s lengths set apart, we have to follow some criteria to represent a closed shape (not only plot ten numbers randomly, but we still have unlimited variants). Changing the length set so we can represent distinct objects, but they still have the same angle sequences. Figure 3.15 shows two distinct shapes with the same angular sequence, but different lengths. The rotations, scales and positions can be

changed, as presented in the figure, but the angular representation is the same. In this case, two distinct turning functions have the same $\theta(s)$, i.e., the same angles in the same sequence.
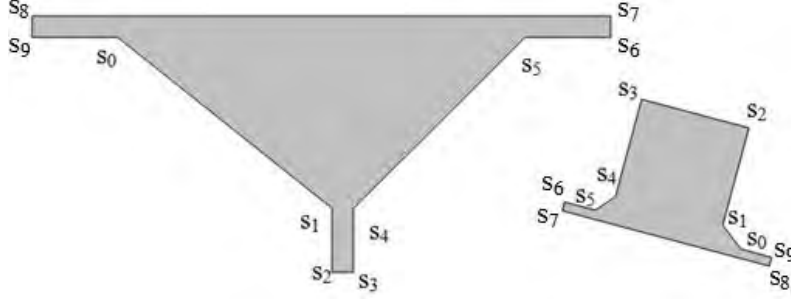
Figure 3.15 - Objects with the same turning function's angles.

Notice that in Figure 3.15 the two distinct shapes with different orientation have the same $\theta(s)$ representation because from the starting point all sequence is coincident. Changes in orientation occur in two ways: the very first angle of the turning function defines the rotation for deviation functions (i.e., what we simply call a turning function in this work), or an angle amount added to each angle of the tangent angle functions (i.e., where the angles represent the angle between a reference angle and the tangent of the shape on that vertex).

A way to quantify the amount of differences between two shapes is the distance between two shapes, as defined by Arkin et al. (1991), an improvement of a proposal of Wolfson (1990). The distance between two normalized shapes $\theta_A(s)$ and $\theta_B(s)$. with $\theta(s) : [0,1] \rightarrow [0, 2\pi[$. Equation (3.13) defines a distance between two shapes A and B:

$$D_p^{A,B}(\sigma, \tau) = \left(\min_{\tau \in \Re, \ \sigma \in [0,1]} \int_0^1 |\theta_A(s+\sigma) - \theta_B(s) + \tau|^p \, ds \right)^{\frac{1}{p}} \quad (3.13)$$

$$d_p(A, B) = \left(\min_{\tau \in \Re, \ \sigma \in [0,1]} D_p^{A,B}(\sigma, \tau) \right)^{\frac{1}{p}} \quad (3.14)$$

where τ is a constant angle to be found in the function minimization and makes the turning function represented in the ring-torus shaped space – presented back in Figure 3.12 – to slide in the poloidal direction, σ is a constant length to be added to every length of one of the shapes being considered, making the shape to slide in the toroidal direction, and p is a constant associated to the distance metric. The maximization of the parameters in poloidal and toroidal direction (σ, τ) is an

adjustment to find the lower value of distance.

3.6.7 Representing circles

On a tangent angle representation – the one that belongs to the turning function – the circle is a straight line from 0 to 2π, or a discontinued line from α to 2π and then from 0 to α. The second case differs from the first by the origin point moved from 0 angle to a generic α.

Considering the turning function as discrete on only the length dimension, a circle is represented by a step function (or a stair function).

If the turning function is discrete on both length and angle axes, what happens to the chain code will also occur: the circle is very hard to have a visual identification, with sequences of alternating values, as shown in Figure 3.16. This figure approximates the way in which turning function is handled here. The left column represent the starting point at the bottom of the circle, starting with the tangent being zero. On the right, the starting point of the curve is the tangent of α.

The same representation that occurs with circles occur with circle arc, but the arc is not complete from 0 to 2π or from α to α. In a complex shape, the circle arc is found by a sloped line or a regular stepped stair function. If the slope is positive, the stair is going up and the arc is outside a convex shape. Instead, if the slope is negative, the stair is going down and the arc is internal to a convex shape, resulting in a concave shape or a non convex shape.

A circle in the turning function is a regular polygon in the tangent angle representation. The only difference between a circle and a regular polygon is that the circle should have the narrow side as possible.
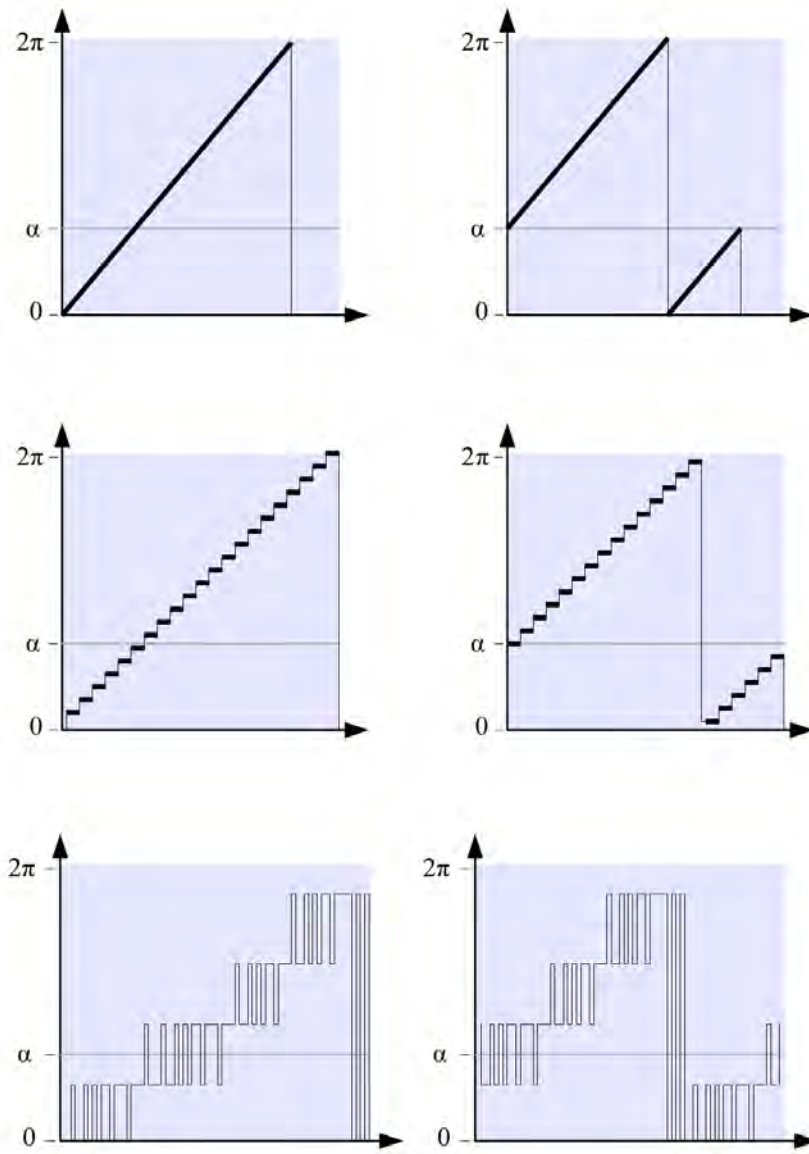
Figure 3.16 - Circle represented by tangent angle function, turning function and chain code. The left column represents the starting point at zero and the right column is the initial a point at α. The first row is the tangent angle.

3.6.8 Bounding box

A dashed line in Figure 3.17 represent a shape with its desired bounding box graphically indicated (a rectangle in gray). We can find mathematically the dimensions d_1, d_2, d_3, d_4, d_5 and d_6 and then we already have $d_{1x} = -d_6$ and $d_{2x} = d_6$. Finding these values is enough to define a bounding box. It can be seen in the figure that from

the starting point (the place marked with an "O") it is possible to find any corner of the bounding box. This is done by running over the shape in the turning function plot (right side) and considering that the opposite direction (e.g., for $d_{1x} = -d_6$) the distance depicted as an "arrow" will be shifted vertically by π. Considering the first angle to be zero, the decomposition of the shape can define the desired bounding box with axis in horizontal and vertical directions.
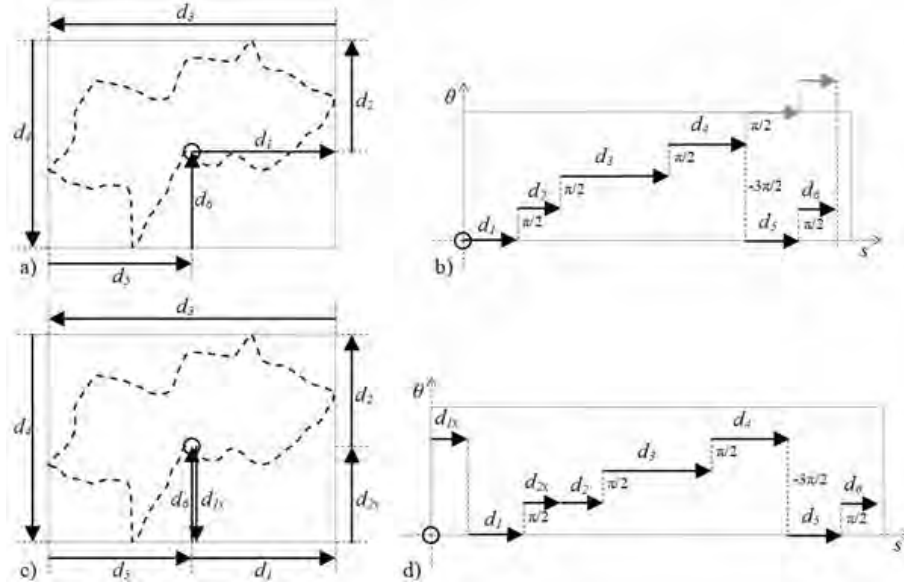
Figure 3.17 - Bounding box in reference direction.

To define a bounding box and keep the starting position, with this reference direction, we must break down the figure in sine and cosine components as shown in the figure below. This will result in a turning function with 4 to 7 segments, depending on the case. It will be less than 7 segments if the starting point is the minimum or maximum in one of the directions defined by the reference axes, in this case, horizontal and vertical. The bounding box for Figure 3.17 have 7 segments. The other extreme would be another shape that matched the starting angle with one of the four bounding box vertices and would thus have 4 segments.

The given shape is represented by a dotted line and its corresponding turning function at right. The small circle indicates the starting point and is a reference for reconstructions proposed on this thesis. To find the bounding box is easy and has the appearance showed in b), where d_5, instead of $\pi/2$ higher in function, is represented by the congruent angle $-3\pi/2$ lower. The bounding box c) result from the turning function presented in d), where two vectors with the same magnitude and opposite

directions are added, identified graphically by d_{1x} and d_{2x}, where $d_{1x} = d_{2x} = |\mathbf{d_6}|$. The sum of the projections of all vectors of a contour in the four basic directions, right, up, left and down, results in a rectangle.
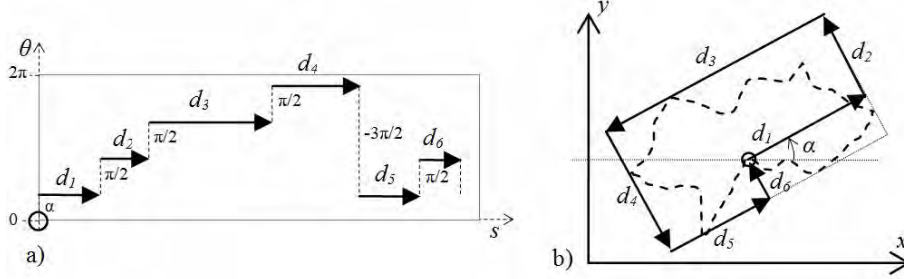
Figure 3.18 - Bounding box in a rotated direction.

Figure 3.18 shows a bounding box rotated by an α angle and b) represents the geometry of the decomposed vectors in α, where the bounding box is identified. The new perimeter is the rectangle perimeter. Using the same way discussed before it is possible to define this curved bounding box.

3.6.9 Operations

In the turning function space each generic segment can be decomposed into lower (s_-) and higher (s_+) vector. To understand this representation consider Figure 3.19, where on the left the trigonometric circle with four quadrants Q_1, Q_2, Q_3, and Q_4 is shown and on its right side the turning function space is represented. A consistent nomenclature is used on both charts.

Consider the dashed line in the right side chart have six thick lines. These two sets of thick lines are decompositions of the two centered thick lines. The leftmost is being named Δs_i and the rightmost is Δs_j. The decomposition that we are proposing here is s_- for the lower thick line and s_+ for the higher thick line. In the example, for Δs_i the lower is $s_- = s_i \cdot \cos(\theta_i - \pi)$.

In this notation we have $s_+ - s_- = \pi/2$ and θ'_i is $\theta_i - \alpha$, where α is the angle corresponding to s_-. We have $\alpha = \alpha_0, \alpha_0 + \pi/2, \alpha_0 + \pi, \alpha_0 + 3\pi/2$ and $0 \leq \alpha_0 < \pi/2$. When using the original reference orientation without rotation then $\alpha_0 = 0$. It is possible always to use the new reference first quadrant corresponding to the vector or part of the vector and have always positive values of sine and cosine. The direction in this case will be defined by the axis direction. If AB in equation 5 is a
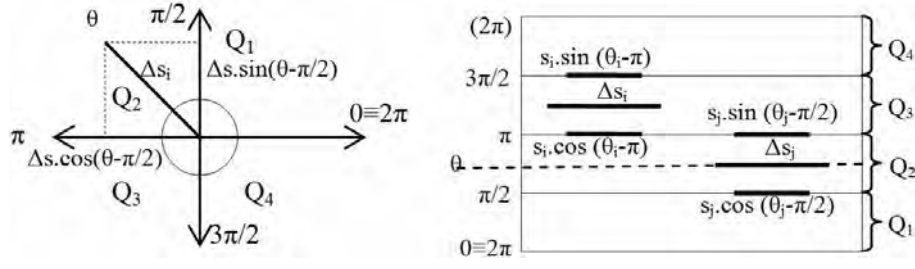
Figure 3.19 - Decomposition on turning function 2-D space.

line, θ_i is constant over AB, and then we have equation 6.

$$s_- = \sum_{i=A}^{B} \Delta s_i \cos \theta'_i \quad \text{and} \quad s_+ = \sum_{i=A}^{B} \Delta s_i \sin \theta'_i \qquad (3.15)$$

$$s_- = \Delta s \cdot \cos \theta'_{AB} \quad \text{and} \quad s_+ = \Delta s \cdot \sin \theta'_{AB} \quad \text{if} \ \ AB \ \ \text{is a line} \qquad (3.16)$$

3.7 Shape analysis assessment

Indexes can be used for spatial pattern analysis for segmented or classified images (MCGARIGAL et al., 2002) to analyze structure of the landscape on the image. The use of indexes allows the categorization of shapes (i.e., separating groups of the same category) for the object-based refinement during segmentation process. Such categorization can be made by matching the best shape fit by the use of metrics and indexes.

A random example would be electing one of the indexes in increments of 20%. However, it is more appropriate to find the most suitable parameters in a controlled manner as a supervised process, by training sets of shapes. This procedure could also be done in conjunction with values, in a combined approach, obtained by the use of image classification of multiple spectral bands, synthetically obtained channels and texture information.

The perimeter $\mathfrak{P} = \sum_{i=1}^{n} s_i$ may be used as a comparative index, and the indices on equations (3.19), (3.20), (3.21) and (3.22) were developed to be more robust using turning functions, where s_i is the side i of the polygon, $\Delta\theta_i$ is the external angle of the ith polygon vertex, θ_i is the angle between the ith side and a reference direction, q is a fixed parameter ($q \in \mathbb{N}^*$) and n is the number of elements (sides or vertices)
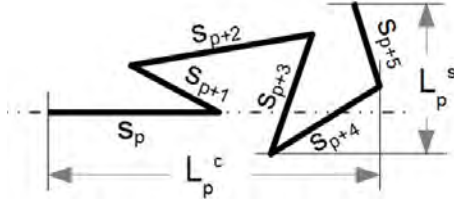
Figure 3.20 - Sides of a rectangle equivalent of a curve segment.
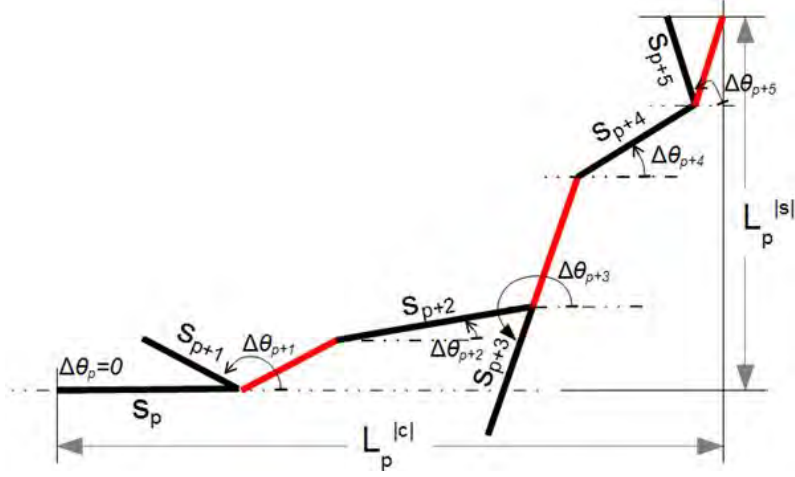
Figure 3.21 - Sides of a rectangle equivalent of a curve segment.

of the polygon.

$$L_p^s \triangleq \sum_{i=p}^{p+q} s_i \cdot \sin(\theta_i - \theta_p) \tag{3.17}$$

$$L_p^{|s|} \triangleq \sum_{i=p}^{p+q} s_i \cdot |\sin(\theta_i - \theta_p)| \tag{3.18}$$

$$\mathfrak{I}_1 \triangleq \left(\frac{\sum |\Delta\theta|}{\mathfrak{P} \cdot n \cdot \pi} \right) \tag{3.19}$$

$$\mathfrak{I}_2 \triangleq \left(\frac{\sum |\sin(\Delta\theta)|}{\mathfrak{P} \cdot n \cdot \pi} \right) \tag{3.20}$$

$$\mathfrak{I}_3 \triangleq \sum_{p=1}^{n} \left(\sqrt[k]{(L_p^s)^k + (L_p^c)^k} \cdot \frac{\displaystyle\sum_{j=p}^{p+q} |\theta_j - \theta_i|}{\displaystyle\sum_{j=p}^{p+q} s_j \cdot n \cdot \pi} \right) \tag{3.21}$$

$$\mathfrak{I}_4 \triangleq \sum_{p=1}^{n} \left(\sqrt[k]{(L_p^{|s|})^k + (L_p^{|c|})^k} \cdot \frac{\displaystyle\sum_{j=p}^{p+q} |\theta_j - \theta_i|}{\displaystyle\sum_{j=p}^{p+q} s_j \cdot n \cdot \pi} \right) \tag{3.22}$$

Based on shape indexes it is possible to compare or select image segments to modify or to control the extent of modification on modeling (VOLOTAO et al., 2011).

In the case of multiple polygons to be compared, the index (3.23), presented by (COELHO et al., 2009), can be used to assess the degree of similarity and to detect dissimilarities, but unlike the turning functions, to use this equation it is imperative to consider the geographic position, including rotation, unlike all other equations presented.

$$\mathfrak{I}_{CSI}(P_A, \ldots, P_n) = 100 \cdot \frac{area(P_A \cap \cdots \cap P_n)}{area(P_A \cup \cdots \cup P_n)} \tag{3.23}$$

3.8 Development and implementation of SegmenTHIS with shape analysis

To carry out and validate the proposed theory presented in this thesis it was necessary to produce a prototype to allow the execution of the proposed steps and to verify the feasibility of the object-based approach. The prototype is being called "SegmenTHIS"[4] was implemented in IDL and based on a prototype used in the work of hue segmentation used by Souto (2000).

After the period of development of this thesis, SegmenTHIS experimental prototype reached the current configuration, containing now the size presented in Table 3.4.

Table 3.4 - SegmenTHIS (prototype developed)

Number of Lines	Functions	Module description
2400	30	main module
2140	29	main module (other functions)
1690	50	shape analysis module
380	18	advanced module for sets (auxiliary)
165	2	main module

A diagram of the main modules of the implemented algorithm is shown in Figure 3.22. The starting point is the hue-based segmentation algorithm, based on Souto (2000).

Modifications in the basic algorithm was made to get better control on the shapes produced. The hue-segmentation method implemented in IDL language is being called SegmenTHIS and allows to change the shape, maintaining its basic characteristics. New features were added to SegmenTHIS and are being described throughout the text. New ways to obtain hue channel from multispectral bands (synthetic hue methods) were integrated to the process, aiming to the use of all multispectral information available and to further improve the results.

The size of the boxes in Figure 3.22 gives a relative notion of the complexity and number of features available in each module. The chain code is a module that attaches to the shape analysis module. The shape analysis module has a large range of

---

[4] SegmenTHIS stands for: segmentation using H, I and S channels. Notice that the last "t" and the "HIS" form the word "THIS", written in uppercase letters to highlight the object-based approach.
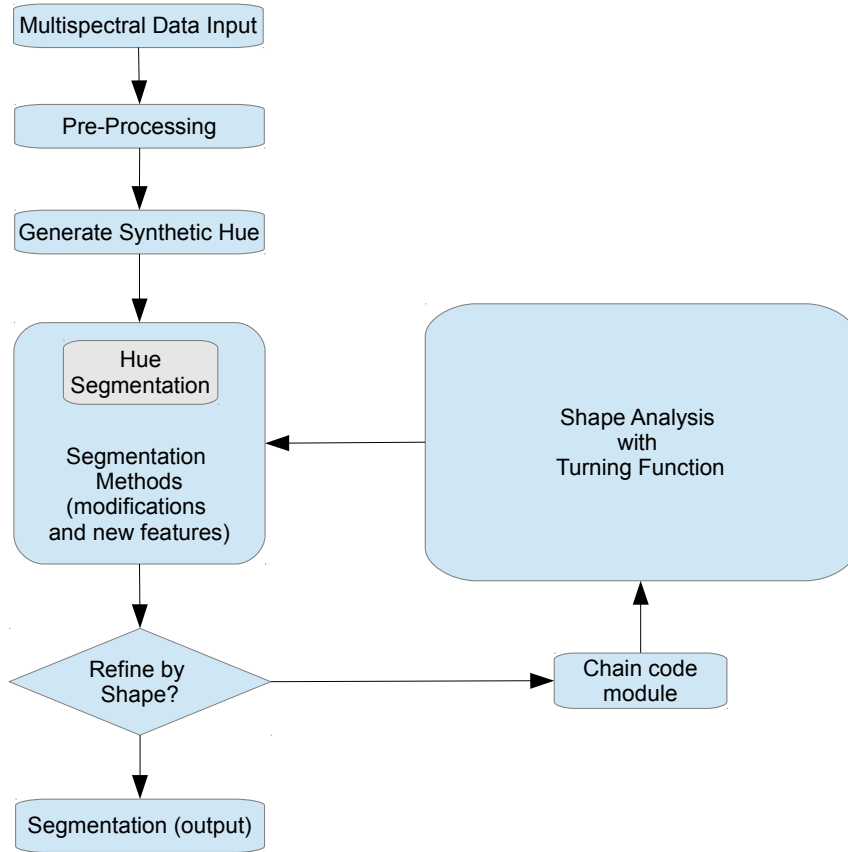
Figure 3.22 - Algorithm overview diagram.

functions available for the analysis of the shapes of the segments, and is specifically focused on the turning function representation.

The pre-processing and generation of synthetic hue images are described in Sections 3.4 and 3.5, respectively. This section describes the modules presented in Figure 3.22.

A sketch of the object-based segmentation kernel is shown in Figure 3.23, a sequence embedded in the loop of Figure 3.22. Figure 3.23 illustrates the relationship between the core SegmenTHIS algorithm just starting on the segmentation stage up to the acquisition of the vector data, represented by the discontinuous line of division in the figure. This figure brings out five basic outcome stages on the kernel of the object-based approach: (a) the hue image used to demonstrate this thesis, (b) the ID image, a segmented image with the segments labeled by numerical coding; (c) identification of foreground by shape (although it is possible to use any attributes); (d) identification of the best matching shapes; (e) remodeled shapes of the object. The shape analysis procedure proposed occurs after the chain code module of Figure 3.22 and
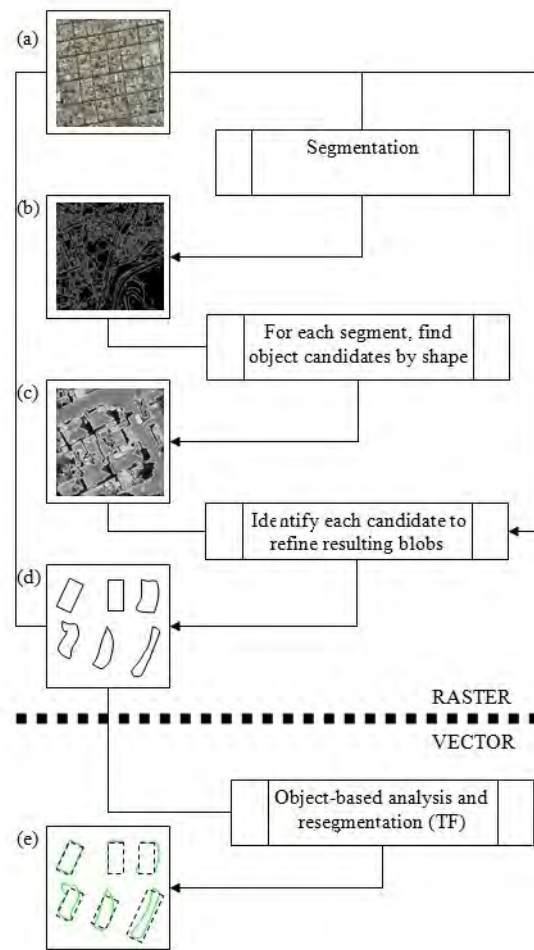
Figure 3.23 - Overview of Object-Based Segmentation: (a) image; (b) ID image; (c) foreground (recognized); (d) best matching shapes; (e) remodeled shapes.

uses vector representation instead of raster.

3.8.1 Pre-processing

To make it possible to achieve better results, the core algorithm was modified several times, in addition to the added modules to the whole proposed system, and the most important are being presented. Figure 3.24 present a schema of the input added processes, as an image pre-processing stage. The input can be submitted to a normalization procedure (in green the added processes), where the data can be histogram-stretched or normalized, by using a given mean and a standard deviation. If the user want normalization without parameters, the default values, considering the byte format, are mean $\bar{f} = 127$ and standard deviation $\sigma_s = 45$. These pre-processing steps makes very significant changes in the results, depending on the input images and the parameters.
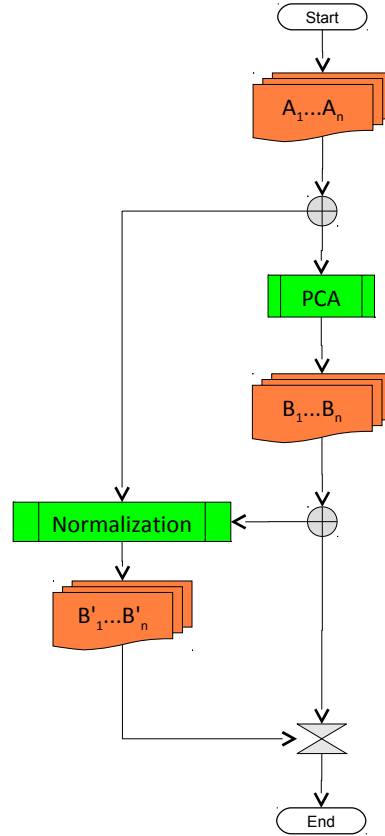
Figure 3.24 - Segmentation overview: pre-processing of multispectral bands.

The second option is the use of a principal component analysis (PCA) transform, a rotation in the feature space that provides the largest possible variance to the

first component. PCA rotates the pixel values on the feature space around their mean so that the principal components are aligned, reducing the correlation between the principal component (PC) channels. Adjacent bands in remote sensing images are often highly correlated, so the last channel PC_N usually contain almost no information and have very low contrast. The rotation in feature space did not change the information, but made changes to the way the data is being represented.

The PC_i channels can be normalized to the default and then a new rotation can get the image back again in bands in RGB space. This way to normalize tend to produce a hyper-spherical data distribution in the feature space, instead of the hyper-ellipsoidal-shaped distribution (for bands with normal distribution). This operation is usually known as decorrelation stretch. This is a solution to avoid low saturation areas.

3.8.2 Segmentation module

Section 2.2 discussed several theoretical aspects about achieving the segmentation from an image. Here and in the next sections, the main aspects of the solution used to make segmentation of multispectral images using color will be presented.

The basic modules of the implemented segmentation algorithm, presented in Figure 3.22, describe the modules developed in SegmenTHIS. Hue-segmentation core module is dedicated to multispectral images, wherein hue can give some benefit (consider the aspects discussed in Section 2.2.1.

Figure 3.25 is a diagram of the segmentation module. Considering $A_1, A_2, \ldots, A_N$ the N original multispectral bands as input, the next step is to define the method to be used: "Moik" or "S-Weighted".

Moik method is the determination of hue of multispectral images by the use of the equations proposed by Moik (1980) and presented in Section 3.4.1, to obtain a synthetic hue.

S-Weighted method is a method being proposed in this thesis to obtain a new hue channel by weighting the saturation bands of all the combinations of three bands.

Sin-Cos is a simple heuristic method proposed to be used to segment the hue information with any segmentation algorithm (i.e., a non-phase-based method).

Hue method is the phase-based algorithm being used mainly in this thesis. It is the
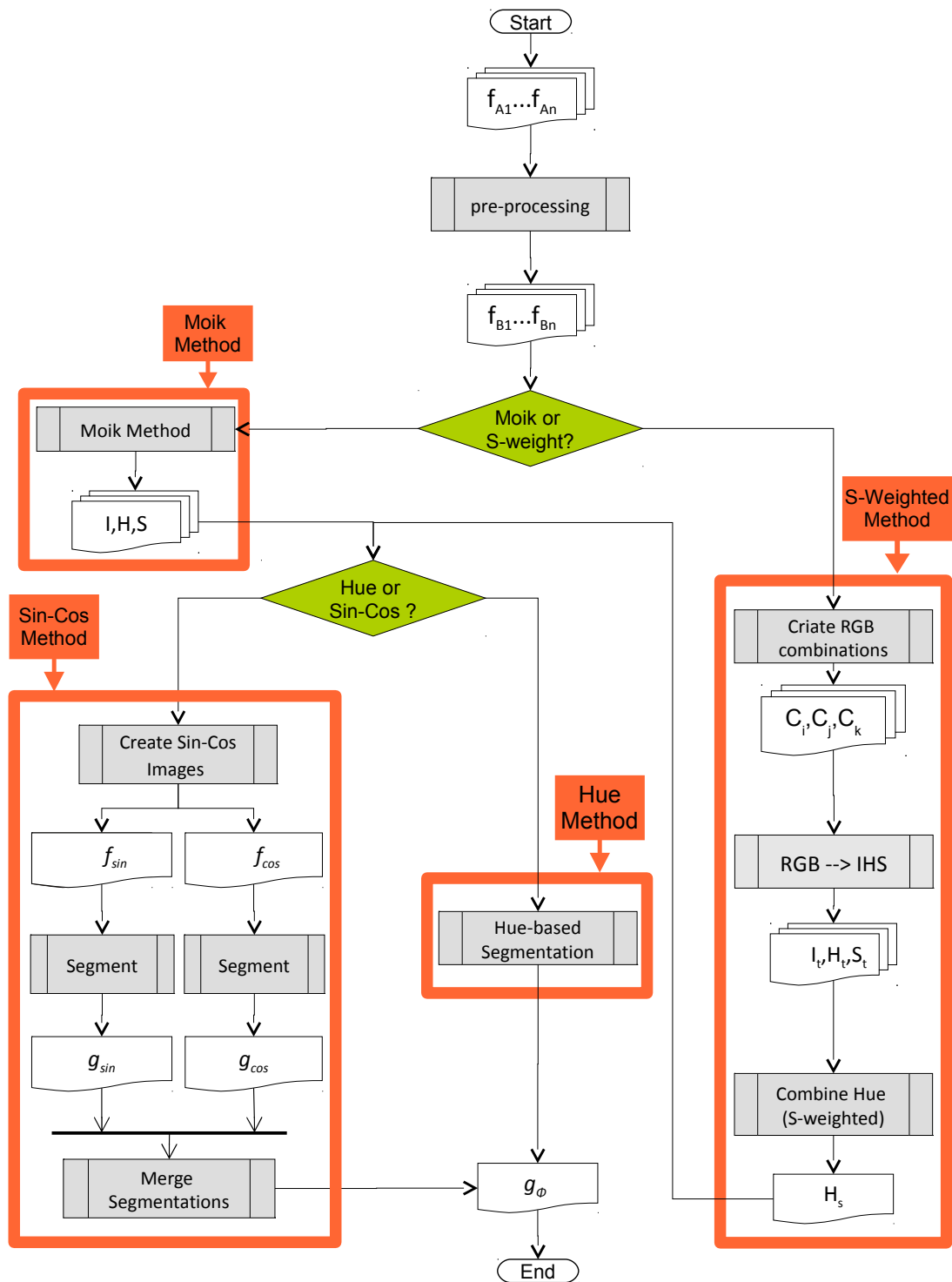
Figure 3.25 - Methods of segmentation implemented in SegmenTHIS.

segmentation algorithm that uses IHS color system to process the segmentation.

The default method uses Moik and Hue methods, where firstly, but after PCA or normalization procedures, the RGB-IHS transform is an internal procedure (not represented in the graphic), producing a single set of I, H and S channels. Having only one phase image the process is to proceed with hue-segmentation, resulting in a segmented image D.

3.8.3 Use of turning function to refine shapes

Figure 3.26 shows a schema to proceed the refinement of image segmentation. This method starts with an image that can be, for instance, a remote sensing raw image of an urban area. Urban areas have many categories of objects with strongly regular geometry.
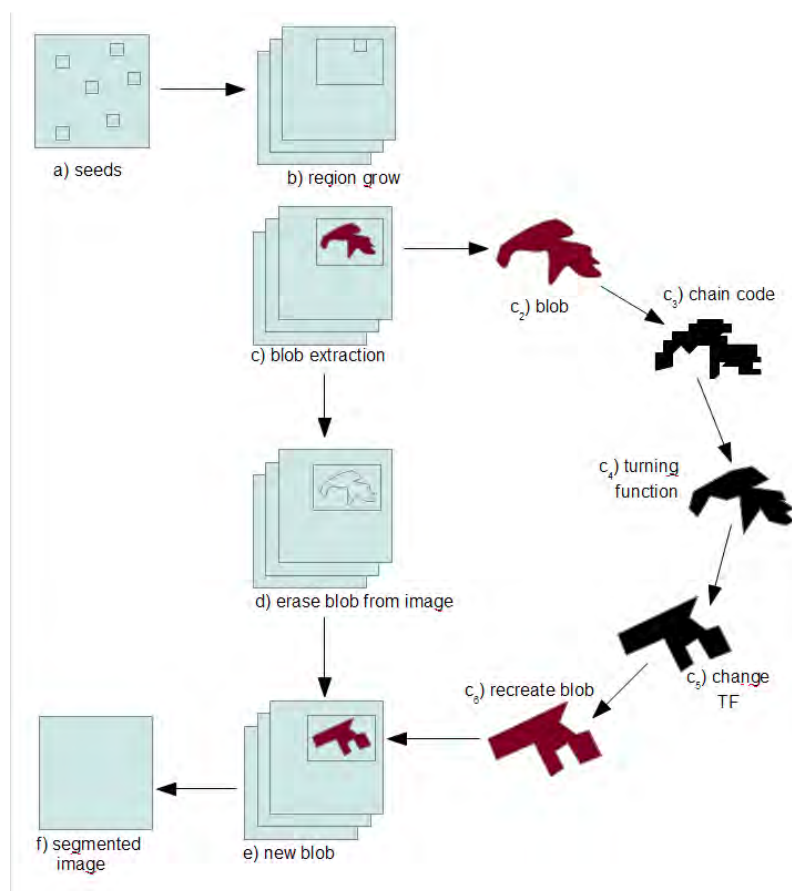
Figure 3.26 - Changing the segmentation based on blob structural rules.

Considering the figure, the first procedure is designed to mark seeds (a) in base image. After having a seed for a blob candidate, the next stage is the region grow algorithm (b). The seeds will be treated as distinct cases, and this is the reason why multiple images have been represented from (b) step. The region grow can be substituted by another process with the purpose of defining an initial blob. After the growth of the region, the blob is extracted (c) and the image (or the blob mask) will be erased (d).

The blob extracted takes a new direction in the figure, considering only the foreground as the blob (c_2), even if there is another blob in the same bounding box extracted. The border of the blob is detected and a chain code is a possible solution, giving the shape definition within the same pixel accuracy. The pixelization effect on the border of chain code (c_3) is unavoidable in a direct process of detection, but converting chain code to turning function (c_4) allows the shape suit any desired format. This makes the turning function very flexible and advantageous to work.

From (c_4) to (c_5) there is a wide open field, where there are many different ways (methods and even approaches) to modify the shape. A simple example is the simplification of the shape by reducing the number of points of the turning function. A more sophisticated example is the use of an object modeling plan. A plain example for this "more sophisticated" case is to work with rectangular roofs and slabs. This will impose the shape to have only four sides. If the blob is not compatible at the segmentation level achieved, nothing can be done and the next blob is processed or a new level of segmentation must be done.

The turning function is manipulated in order to achieve the requested properties and then superimposed on the same reference position and orientation of the enlarged bounding box, that have to be put back in the original place (but now with a new updated shape). The new blob is redrawn (e) and the segmentation will need to incorporate the new blobs (f). While the blobs does not need to take the entire image, there is no guarantee that the blob will have no intersection with the surrounded region. This means that it is necessary to deal with overlaps. By the definition of segmentation the overlap is not allowed, and this becomes a new issue. One solution to overlap is to superimpose blob over any background area and define in the intersection a new segment, making a kind of over-segmentation only on the borders of objects which overlaps.

Figure 3.26 gives only a picture of the object-based purpose, but Figure 3.27 presents a lower level of the interactive shape analysis algorithm as implemented in the ex-
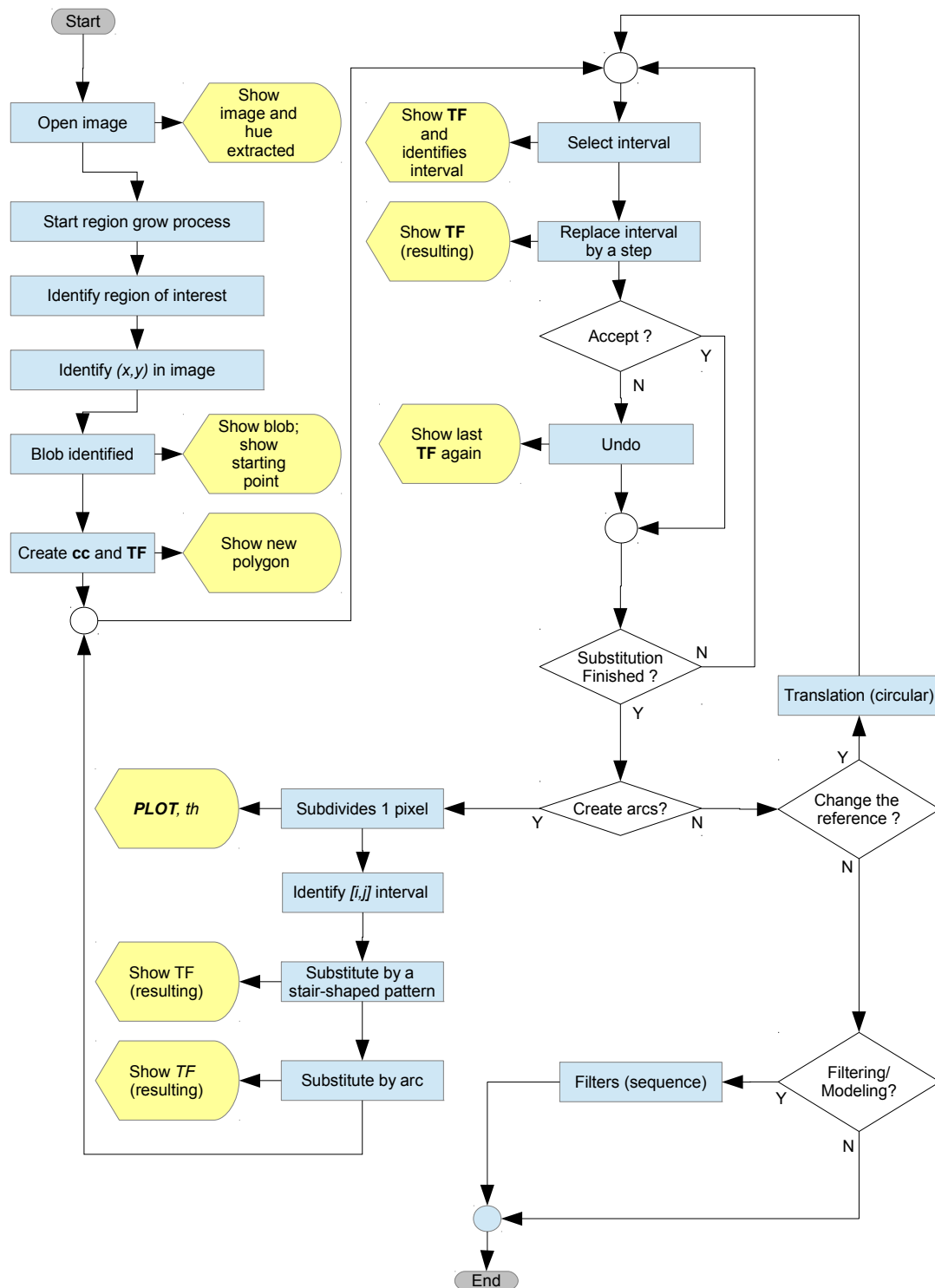
Figure 3.27 - Shape analysis interactive module.

65

perimental prototype.

The image opening procedure, boosted to allow a greater number of image types (e.g., PNG, JPEG and images up to 24-bits), is the starting point of SegmenTHIS. In the same time the original image and the hue is automatically displayed. After setting all the required parameters present in the graphical interface, the region-growing algorithm starts by finding one-by-one the segments. But if the user first selected the manual option of chosing blobs, then the system waits for the identification of a point in the image to be the seed for the blob. If the growing segment meets the parametrized specifications then blob is shown, the origin of the contour extraction is presented and the chain code module generates the coding. The chain code is then converted into turning function and a graphic showing the blob, the curve and the turning function is shown.

The algorithm waits for the user to identify the interval by cursor over the turning function displayed to be replaced by a line and then makes a temporary substitution, changing the turning function length due to the substitution. If the user is satisfied by the results, accept the modification, else the turning function returns to the old shape. This continues until the user finishes the step by step substitution.

The module of creation of arcs first substitutes the turning function by a new turning function with all elements subdivided into "one pixel" width to allow long segments to be subdivided if required. The user visually identifies the interval to be replaced by the arc, using the displayed chart, and the algorithm automatically replaces the interval by a well-behaved stair-shaped pattern that fits the old pattern interval and tries to match the starting and ending angles. After showing the results, the step module returns, allowing a new cicle of changes.

Finishing the line and arc operations, the user can change the reference (by slides in poloidal or toroidal directions by a default multiplier) and can continue with a new point of view (the slide don't change the shape).

The last option is a filtering and modeling option, implemented in a very basic form. All operations in this version of SegmenTHIS allows filtering, modifying angles or lengths by a factor and using the geometric constraint (i.e., making a correspondence between the turning function and the 2-D shape in euclidean space). Some functions are available and are useful to create new procedures, such as the substitution of an interval of the turning function by a defined pattern (e.g., arc and segment).

3.8.4 From chain code to turning function

From chain code to turning function we make a simple conversion from the codes to distances and angles. This makes chain codes as a subset of turning functions.

The discrete geometry that is possible with a chain code representation is limited. Chain codes can represent only sequences of touching pixels in a path. This path can be opened or closed and it is possible to have crossings. Anything different is not possible at all.

Turning functions can represent points at any distance away from the previous one. The path can be opened or closed and it is possible to have crossings too.

3.8.5 Chain code is a vector representation

Chain code is a way used to represent digitized lines. It is a description of the shape by a relative movement code (PRATT, 2007). The classic Freeman code is the classic eight-element (FREEMAN, 1974) code, but there are variants: a four symbols, a sixteen-neighborhood (MARCHAND-MAILLET; SHARAIHA, 2000) and primitives chain code that "represents either a portion of a zero to three pixel chain, or an endline, junction, or break feature" (O'GORMAN, 1988), a simple way to compact the code.

It is possible to represent connected regions of pixels and represent all lines and polygons defined on a 2D image with this kind of representation. Contour representation by chain code relies on the fact that even in an intricate feature there is always one and only one path around it.

Chain code can be used as a lossless compression algorithm for bi-level images. It encodes each binary segment and keeps the starting coordinate of each segment for the reconstruction.

Freeman chain code of eight directions is a classical way to represent contours, proposed by Freeman (1974). The symbols are the numbers in the set $\{0, 1, 2, 3, 4, 5, 6, 7, 8\}$, each representing one direction. An alternative representation computes only the directions: east, north, west and south for the digit. There are plenty of developed algorithms and methods to do a number of uses.

Chain code and turning function at first glance seems to be different things. Chain code is defined by a string defined by an ASCII sequence with 8 distinct digits, each as a symbol of both direction and preset distance. Turning function, as we are

defining, requires distance and angle to be explicitly specified. This demands two numbers instead of a single symbol.

Algorithm 1: Function to obtain a new segmentation from the intersection of two initial segmentations (as implemented in IDL language).

Data: A and B: two images of segmentation with one band containing ID (long integer)
Result: D: one new segmentation image with new ID labels

$A \leftarrow$ first image;
$B \leftarrow$ second image;
$C \leftarrow A + \max A * B$ (local variable);
$D \leftarrow C$ (local variable to represent each pixel position);
$maxc \leftarrow \max(C)$;
$i \leftarrow 0$;
$j \leftarrow 0$;
$flag \leftarrow 0$;
repeat
    $pos \leftarrow$ **where** $(C = i)$;
    if **$n\_elements$**$(pos) \neq 1$ **then**
    | $flag \leftarrow 1$
    end if
    if $flag \neq 1$ **then**
        if $pos \neq -1$ **then**
        | $flag \leftarrow 1$
        end if
    end if
    if $flag$ **then**
        $D[pos] \leftarrow j$;
        $j++$;
        $flag \leftarrow 0$
    end if
    $i++$;
until $i > maxc$;
return D

Algorithm presented in Figure 1 was implemented in IDL. The function $WHERE()$ results in the array of the solution of the condition inside the parenthesis, if any, or in the number -1 otherwise. The function $N\_ELEMENTS()$ returns the number of elements contained in a given variable.

3.8.6 Shape modification example

The shape analysis module allows modification of shapes and this subsection shows the results of a case of modifying a shape. Before starting the segmentation, the user can mark the shape analysis option and the individual or complete shape modification. The prototype today is fully operational only on the "blob" definition option, which makes the IDL programming environment to wait for the user to mark a position to start the growing of the region.
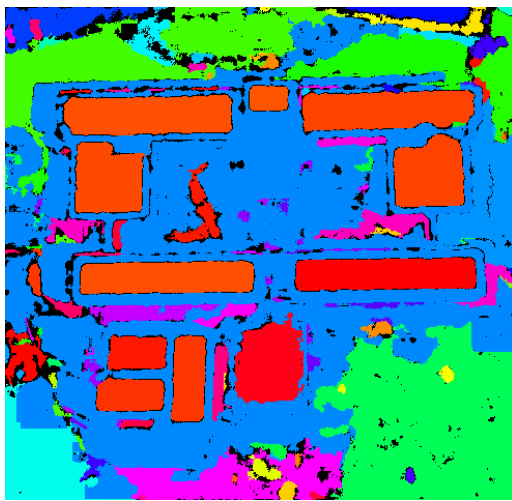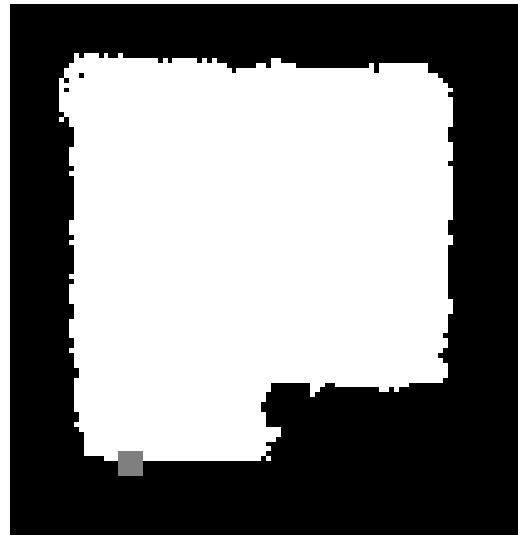
(a) Segmentation   (b) Blob and starting point.

Figure 3.28 - Segment a blob: (a) segmentation; and (b) the blob

After the indicated area is pre-segmented by the defined segmenter parameters the user have to interact with the software in the way described in Figure 3.27. Figures 3.28 shows a segmentation and a blob extracted, where the little gray "dot" is the turning function origin. Figures 3.29 shows a sequence, where the meaning is highlighted in colors to show the visual identification made by the user, who needs to know the consequences in turning function (Figures 3.30(a) to 3.30(d)).

(a)

(b)

(c)

(d)
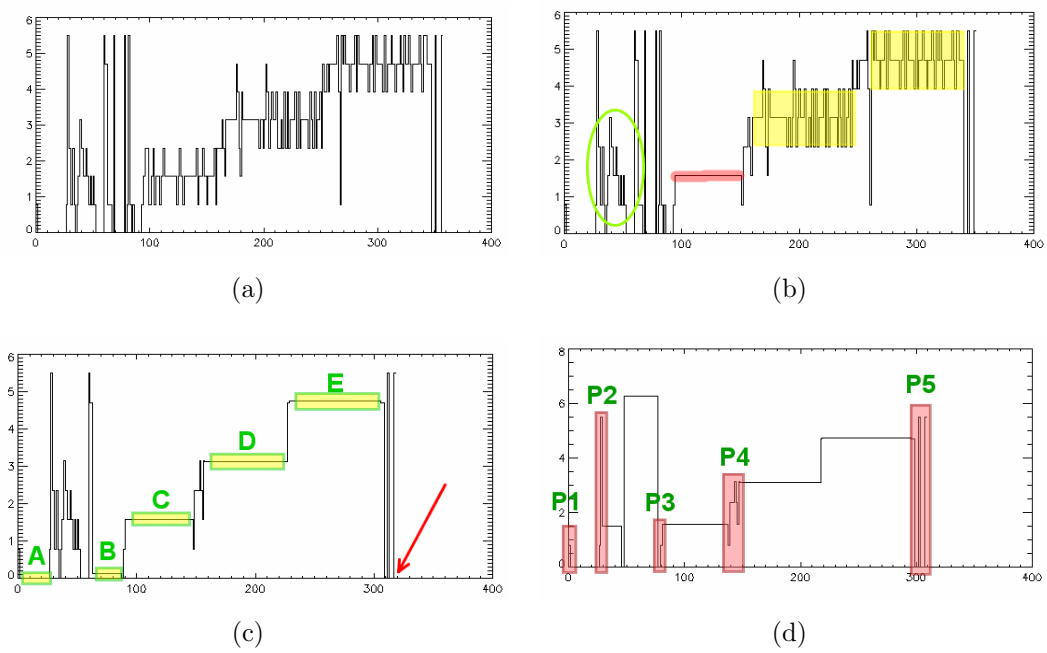
Figure 3.29 - Turning functions: (a) Original TF of roof; (b) Finding irregular lines by turning function analysis; (c) Finding straight lines in shape by turning function analysis; and (d) Finding corresponding vertices on shape by turning function analysis.

70
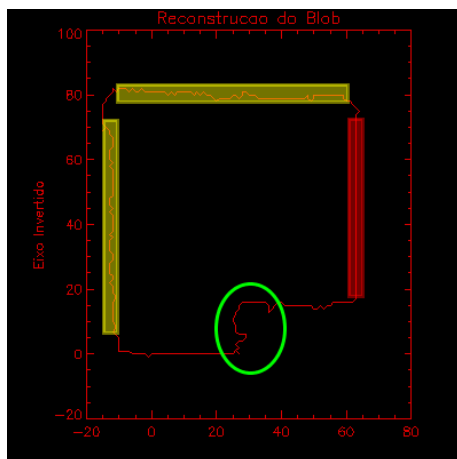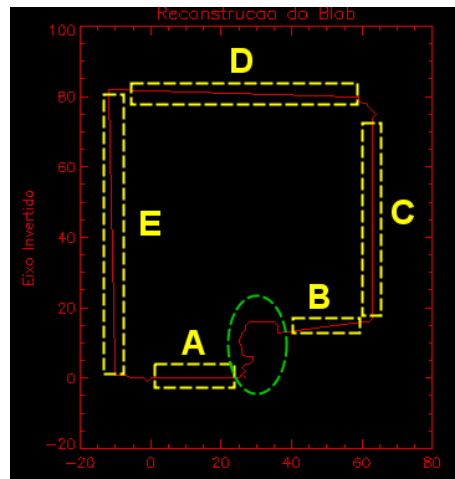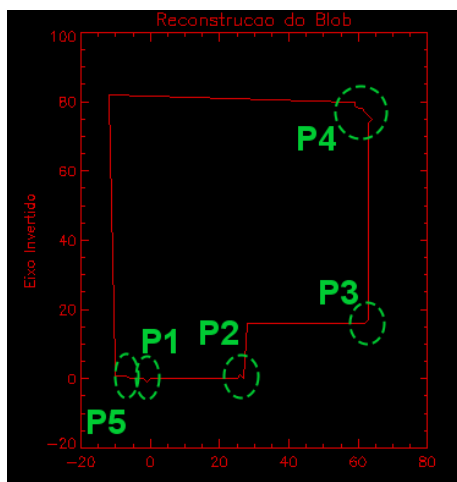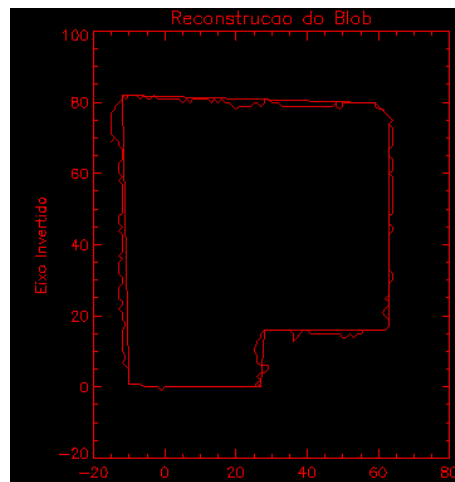
Figure 3.30 - Show what is happening with the shape: (a) identification of lines and patterns; (b) identification of straight lines; (c) identification of corners; and (d) overlapping old and refined shapes.

3.9 Obtaining results

The performance of the proposed methods are tested by comparison to other methods. The test images was based on a real satellite image, classified using ENVI software and then segmented by a connected component labeling algorithm. An IDL prototype was used to create the simulated images from both the classified and the segmented images. Then a RGB-IHS conversion provide new channels and some errors has been introduced in the intensity channel. Figure 3.31 was generated by a
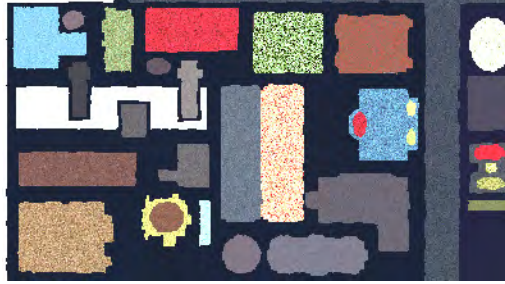
Figure 3.31 - Gaussian simulated image.

synthetic generator with the input of a reference segmentation image file F_R, with connected segments continuously labeled, with sequential ID numbering starting from zero.

In order to create a phantom to produce simulated images for test, a simple polygon drawing, simulating an urban area was drawn (Figure 3.32). The shapes on this image will produce segments.
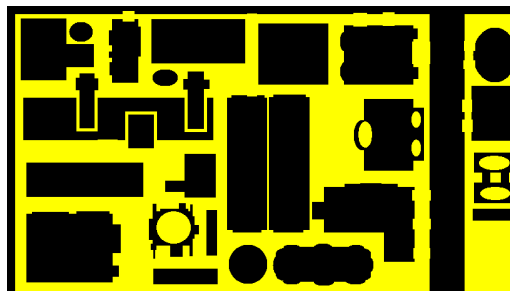
Figure 3.32 - Shape construction to create a phantom image.

The segmentation of Figure 3.32 produces the image shown in Figure 3.33, used

72

as the segmentation reference image. The gray representation maps distinct ID by distinct colors.

(a)	(b)
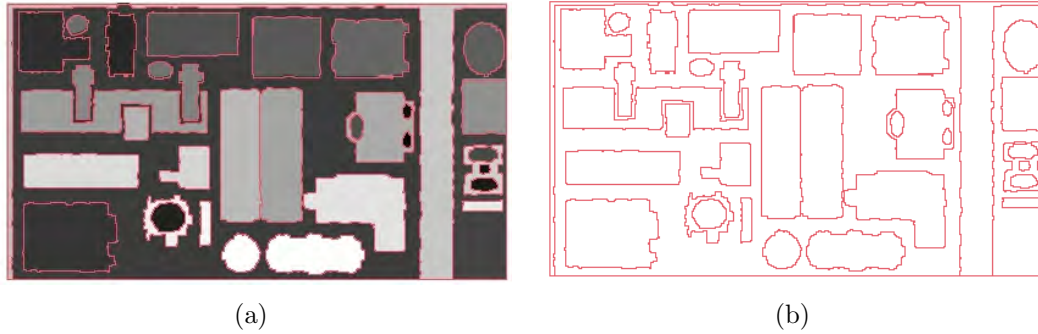
Figure 3.33 - (a) Gray representation of segments of the phantom file and (b) contour.

With this segmentation reference image we defined the number of segments, with every segment corresponding to one ID. The values to fill the segments by a gaussian distribution model depends on samples obtained from an image. The image used to define the regions of interest and the spectral values is Figure 3.34, a Plèiades 431 composite image in Rio de Janeiro area.

Figure 3.34 - Image used to define the regions of interest and collect spectral values.

Figure 3.31 is one of the resulting simulated images. The shape was obtained from the reference phantom (which contours are shown in Figure 3.33) and the specral information was obtained from the spectral information of the regions of interest (ROI), one ID of the segmentation reference (Figure 3.33) for each defined sample region (obtained by the image shown in Figure 3.34) .
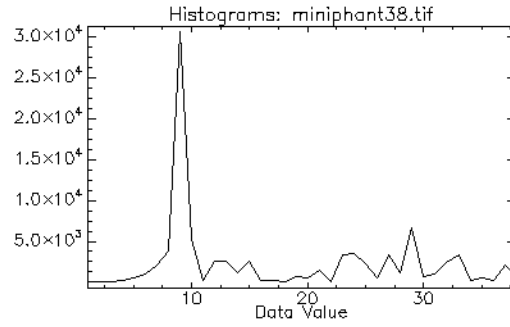
Figure 3.35 - Histogram .

A reference for segmentation used to make the assessment (Figure 3.37(b)) of the results was manually produced in colors and then converted into a phantom image (not segmented but having all connected component labeled in the sequence of natural numbers from the sub-region identifier $ID = 1$ (i.e., the segment numbered "1") to the number of segments.

The histogram of the segmented (phantom) file starts in 1 and ends with the number of segments, showing for each ID number (i.e., for each segment) the number of pixels (Figure 3.36). Thus, the histogram of a segmented image indicate the number of pixels per segment. An image of Recreio dos Bandeirantes, taken by IKONOS

Figure 3.36 - IKONOS image of Jacarepagua area – Rio de Janeiro, RJ, Brazil.

74

satellite, of Rio de Janeiro (Figure 3.36) was selected to be used in the evaluation of the developed segmentation methods because the place was known by the image interpreter. A colored image (fig:1bjac4verdade) was obtained from manual digitizing of the IKONOS image (Figure 3.37(a)). The maximum accuracy is required to avoid errors in the metric evaluation.

(a)   (b)

Figure 3.37 - (a) Area of IKONOS image with 4 bands; (b) Manual identification of segments.

(a)   (b)
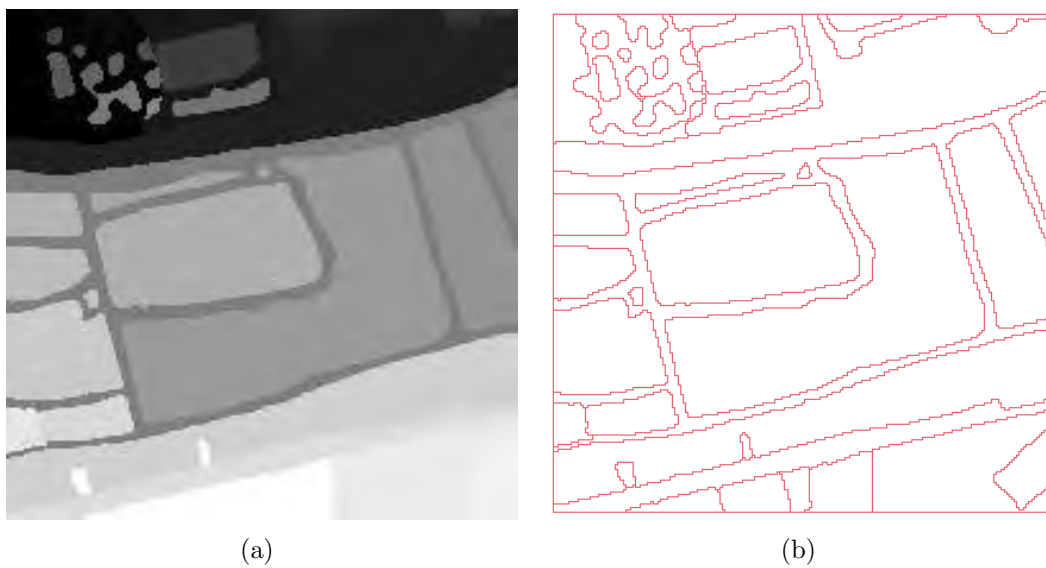
Figure 3.38 - (a) Reference segmentation image, (b) reference contour extracted.

The reference image (Figure 3.37(b)) was segmented to become the "truth" for the tests. By using a segmentation algorithm in a cartoon colored image the result is the change of the pixel values to the image ID, a natural numbering sequence established by the segmenter labeling module. The gray representation (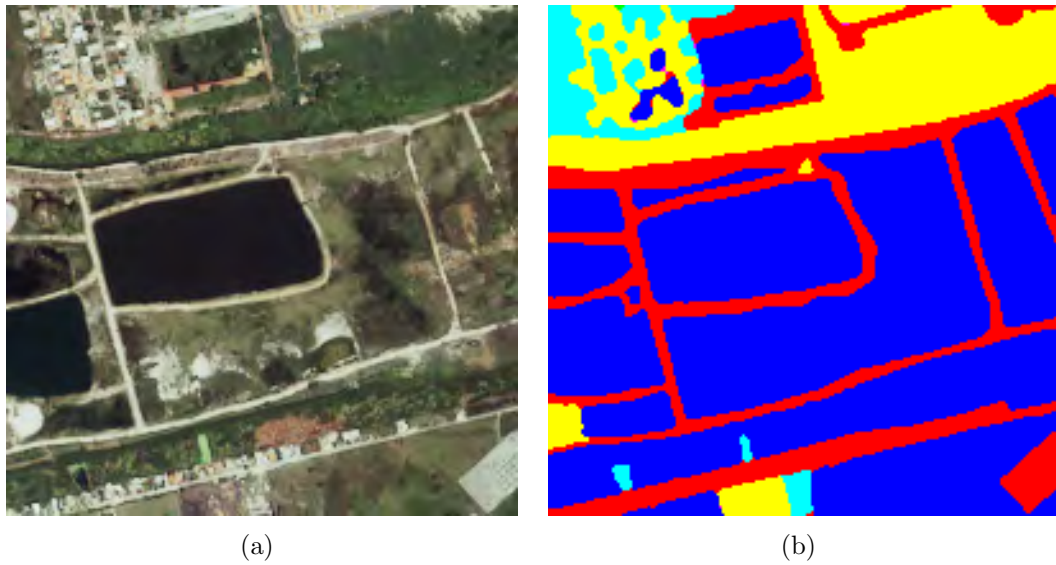Figure 3.38(a)) is the result of a histogram stretching process done for visualization and printing. The reference file have each segment labeled with an ID without visual representation. The contour extracted is shown in Figure 3.38(b).

3.10 Comparing segmentation results

A prototype for evaluation of segmentation was used to calculate the performance metrics described in details by Delves et al. (1992) and Lucca (1998). The measures of performance are used to evaluate the result of segmentation, with the values in the interval $[0, 1]$. A good segmentation will result in a metric close to 1, and bad segmentation causes the metric values to be close to 0.

Table 3.5 - Evaluation metrics

Metric Name	Metric Indicative Meaning
Fitxy	positions of the centroids
Fiti	coincidence in average intensities
Fitn	closeness of sizes
Gform	similarity of shapes
MGlobal	simple average of all scores achieved

The goal is to compare a reference image with the segments generated by the segmenter. The reference image contains the segments used as truth for segmentation. The inputs of the algorithm are:

- Inputs:

    a) original image submitted to segmentation;

    b) ID reference image containing the truth segments;

    c) ID segmented image resultant from segmentation.

- Outputs:

  a) table containing performance measures of all segments related to the reference image;

  b) table summarizing the means and standard deviation of the computed performance.

The use of empirical methods is an indirect way to measure and evaluate algorithms by the use of test images and the quality measurement (ZHANG, 1996). To quantitatively assess the segmentations produced by the proposed methods using SegmenTHIS and to make comparisons there are two kinds of empirical methods:

  a) discrepancy metrics;

  b) quality metrics.

The evaluation of this thesis will be made by the use of methods of discrepancy to compare the results of the proposed segmentation methods. A brief theoretical framework as presented by Lucca (1998) is given hereafter.

This option permit the evaluation of different algorithms and only the input and the output of the segmentation methods are needed and the algorithm itself is not used, making possible to compare distinct methods even when produced in distinct softwares. The discrepancy metrics can be applied to find out different properties of the segmentation.

A digital image is a 2-dimensional function $f(x, y)$, where the support S defines all pixel positions, represented by each point $s \in S$, where $s = (x, y)$ is the pixel position. Considering s the position of the pixel and $f(s)$ is the function that provides the n-dimensional value of the pixel, the pixel is represented by the pair $\{s, f(s)\}$, S is the support grid, and the whole image is represented by the function $f : S \rightarrow \{\nu_1, \nu_2, \cdots, \nu_p\}$, where ν_i is the image value corresponding to the position $s \subset \mathbb{N}$.

Segmentation is a function $g : S \rightarrow \{r_1, r_2, \cdots, r_k\}$ where r_i is the segment with ID=i. Segments are sub-regions of the support S. A segmentation of the grid S for a uniformity predicate P, where $P(r_i)$ =true or $P(r_i)$ =false is a uniform function considered, have the following conditions (GONZALEZ; WOODS, 2010; PAL; PAL, 1993; FU; MUI, 1981):

a) $\displaystyle\bigcup_{i=1}^{k} r_i = S,$

b) r_i is a connected region, $i = 1, 2, \cdots, k,$

c) $r_i \cap r_j = \emptyset, \forall i, j, i \neq j,$

d) $P(r_i)$ =true for $i = 1, 2, \cdots, k$, and

e) $P(r_i \cup r_j)$ =false for $i \neq j$, with r_i adjacent to r_j.

If two regions r_i and r_j are adjacent, then $P(r_i)$ =true, $P(r_j)$ =true and $P(r_i \cup r_j)$ =false.

If a segmentation is represented by g and the support set is $\{s \in S\} : g(s) = r_i$ then the set is defined by $g^{-1}(r_i)$. the size of the sub-region r_i denote $\#g^{-1}(r_i)$. The mean intensity is the average of the radiometry of the sub-region r_i is denoted as $< r_i >$.

When there are two segmentations g_1 and g_2 and the segmentation results are two sets of regions $\{r_1, r_2, \cdots, r_k\}$ and $\{m_1, m_2, \cdots, m_k\}$ respectively, we can compare both by comparing each pair of elements (r_i, m_j).

3.10.1   Shape fit – Gforma

The shape fit is the number of elements of the intersection of two sub-regions divided by the number of the union of two regions. In the ideal case (i.e., when both segmentations are equal) then this index will be $Gf(r_i, m_j) = 1$ for $r_i = m_j$.

$$Gf(r_i, m_j) = \frac{\#g^{-1}(r_i \cap m_j)}{\#g^{-1}(r_i \cup m_j)} \tag{3.24}$$

3.10.2   Position fit – Fitxy

Position fit is shown in Equation (3.25) and is the index for each pair of segments (r_i, m_j) where the value represent the proximity between the regions. If r_i and m_j

have the same geometric center then $Fitxy(r_i, m_j) = 1$.

$$Fitxy(r_i, m_j) = 1 - \frac{xd(r_i, m_j) + yd(r_i, m_j)}{2} \tag{3.25}$$

3.10.3   Intensity fit − Fiti

The intensity fit is shown in equation (3.26) and represent how the mean values of the regions are close. If they are the same this index will return 1.

$$Fiti(r_i, m_j) = 1 - \frac{|< r_i > - < m_j >|}{< r_i > + < m_j >} \tag{3.26}$$

3.10.4   Size fit − Fitn

Size fit (equation (3.27)) compares the difference between each two segments. If the segments have the same size, then $Fitn = 1$, otherwise $Fitn \in [0, 1[$.

$$Fitn(r_i, m_j) = 1 - \frac{\left[\#g_1^{-1}(r_i) - \#g_2^{-1}(m_j) \right]}{\#g_1^{-1}(r_i) + \#g_2^{-1}(m_j)} \tag{3.27}$$

4 EXPERIMENTAL RESULTS

Lucca (1998) compared segmentation results by using a set of four metrics and a global index. The description of segmentation given by Haralick and Shapiro (1985), states that the regions "should be uniform and homogeneous with respect to some characteristics such as gray tone or texture" was considered as well as the assertion that achieving all the desired properties is a complex job.

(a)
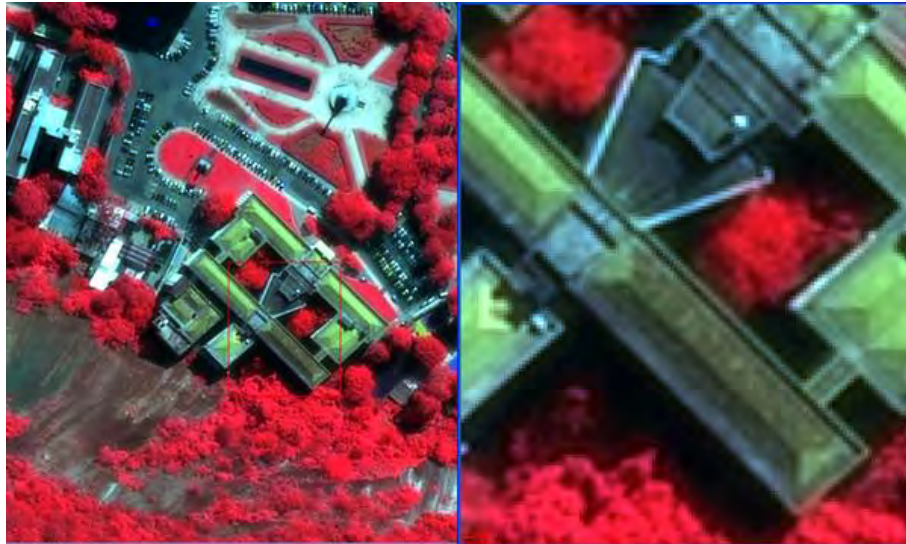
(b)

Figure 4.1 - Color compositions RGB of Pleiades-1 Image: (a) true color composite 321; and (b) false color composite 432.

Strictly uniform and homogeneous regions tends to lead to the creation of ragged boundaries and small holes. The segmentation process can be evaluated by quantitative and qualitative analyzes. Qualitative assessment is based on visual inspection of the shape and number of segments. Quantitative assessment is based on numbers defined by metrics of the image related to the number of segments, mean values, size of segments, number of classes, processing time etc.

Figure 4.1(a) is the real color composite image corresponding to the same image shown in false color on Figure 4.1(b), with a color composition using bands $3-2-1$ for the $R-G-B$ channels. This is one of the images used in testing the methodology, and it is clear that the choice of different bands leads to different results, So, for a number of different cases we need to find similar conditions to make a reasonable comparison, but sometimes this choice is infeasible.

4.1 Evaluation of the IHS segmentation algorithm proposed by Souto and SegmenTHIS

There is a number of segmentation techniques available and each algorithm has its own characteristics. Not all are good for the same set of image and application and there are numerous applications and processes that make use of image segmentation.

The quantification of the performance of a segmentation is a challenging task because of some reasons, including the multiplicity of possible ground truths. There are methods to objectively evaluate the segmentation based on a set of available truths Unnikrishnan et al. (2007). However, as there is no availability of multiple truths, the assessment of the methods proposed in this thesis will be made by comparisons of the results in a simple and objective way.

Figure 4.2 is a Pleiades image of Praia Vermelha area, Rio de Janeiro, RJ, Brazil. Figure 4.22(a) is a color composite of 431 and Figure 4.11 is the resulting hue.

(a) Band 1

(b) Band 2

(c) Band 3
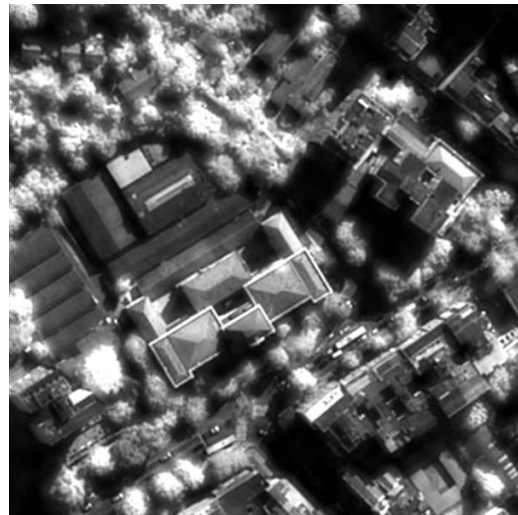
(d) Band 4

Figure 4.2 - Original PLÈIADES image based on fusion of 4 multispectral 2 meters bands with a 0.5 meter panchromatic band of the Praia Vermelha area, Rio de Janeiro, RJ, Brazil.

Figure 4.3 is a result of the SegmenTHIS program for a 0.5 meter very-high resolution Pleiades 1A image. Figure 4.3(b) is a segmentation with a colored color-table. Figure 4.3(c) is a superimposition of the image with the segmentation.

83
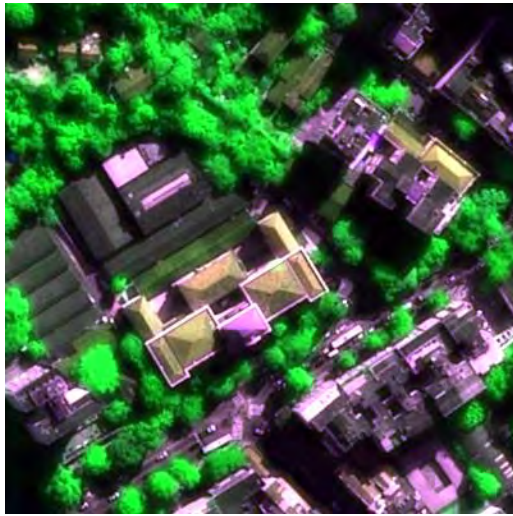
(a) Image of the area, for visual reference, using 342 Pléiades composition
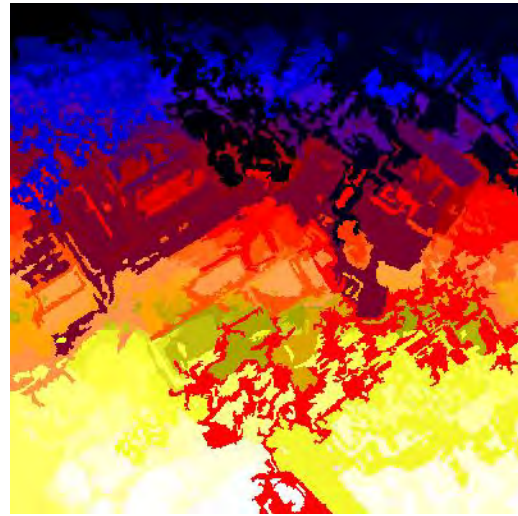
(b) A segmentation to be compared
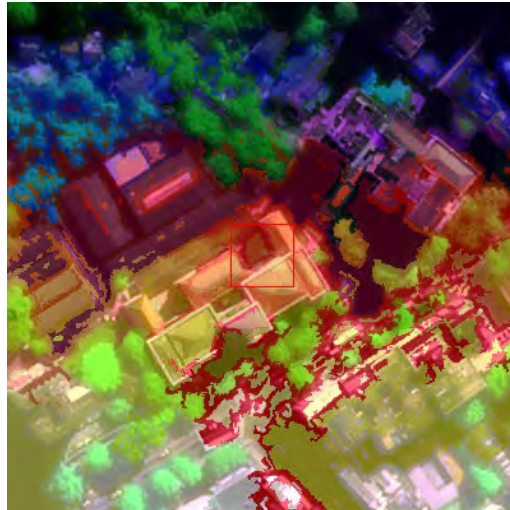
(c) Superimposing image and segmentation

Figure 4.3 - Assessment of the algorithm.

Figure 4.4(a) is a Pleiades image of Rio de Janeiro, with a building of IME. The comparison of the result depends on similar product to be used in comparison, and the segmentation process has many parameters. The following images are some possible results.
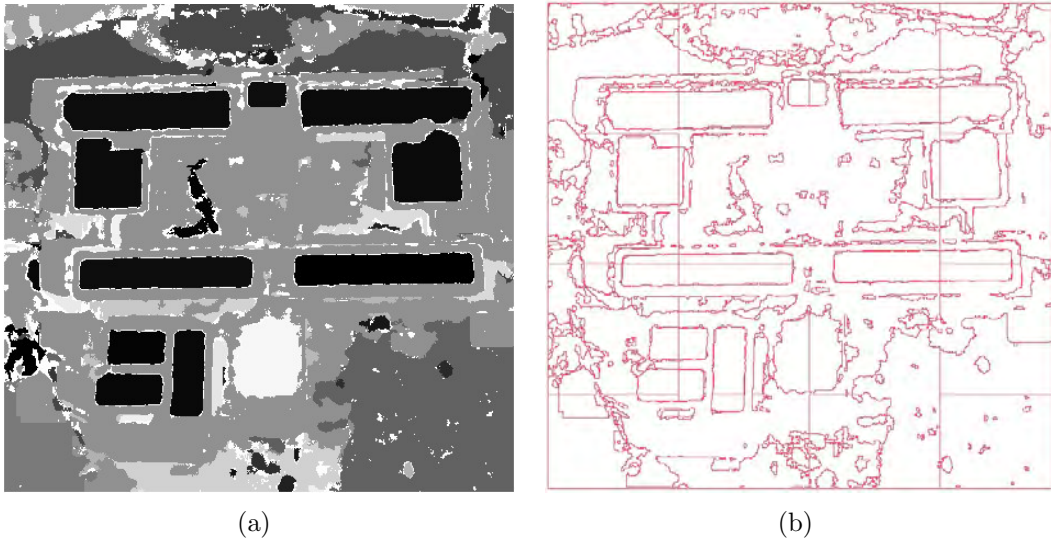
Figure 4.4 - Segmentation SegmenTHIS (threshold=50) (a) output in gray for visualization; and (b) contour identification.
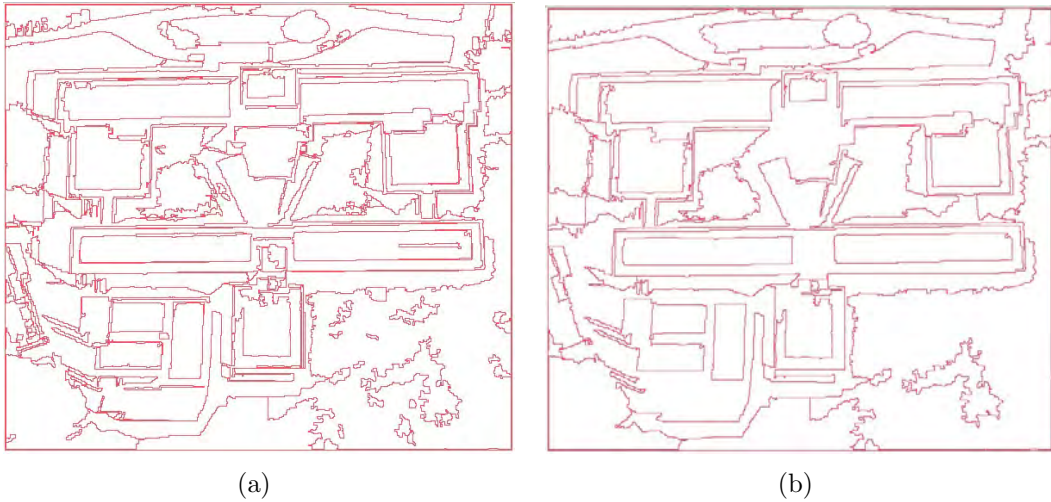
Figure 4.5 - SPRING segmentation results for: (a) $p=(50,30)$; (b) $p=(50,1000)$.

Tests with both SPRING and SegmenTHIS was done with hundreds of combinations was produced. Using some set of parameters produced a bad result. The visual assessment condemned them. To show some examples of this results we present SPRING sample of segmentation outlines in Figure 4.6.

SegmenTHIS segmentations are shown in Figure 4.7, with varying parameters. Figures 4.7(a) and 4.7(b) uses threshold value 90 and the four others, 4.7(c), 4.7(d), 4.7(e) and 4.7(f),
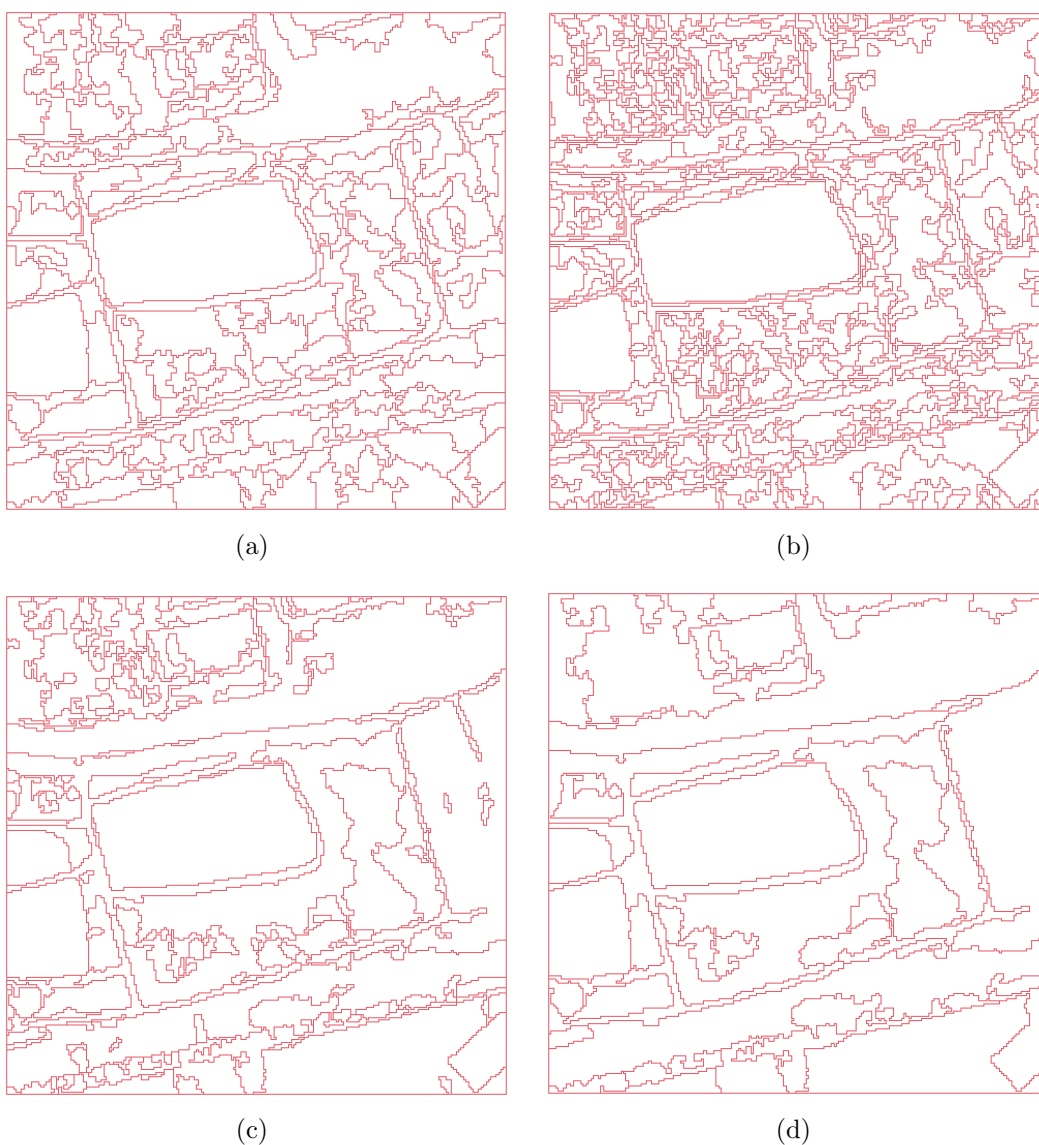
Figure 4.6 - Spring segmentations with (a) p=(30,10), (b) p=(30,80) and (c) p=(60,80).

The comparison of the performance of the different segmentations are dependable on the segmentation method used. The performance of hue-segmenter compared to multispectral region growing was much worse.

Spring is able to detect roads easily while the SegmenTHIS performance in the roads present in this example is very poor. This is due to the nature of the image, the pre-processing and the spectral response, where the mathematically calculated hue is transferring insufficient information to the segmenter.

(a) 90fv4-5
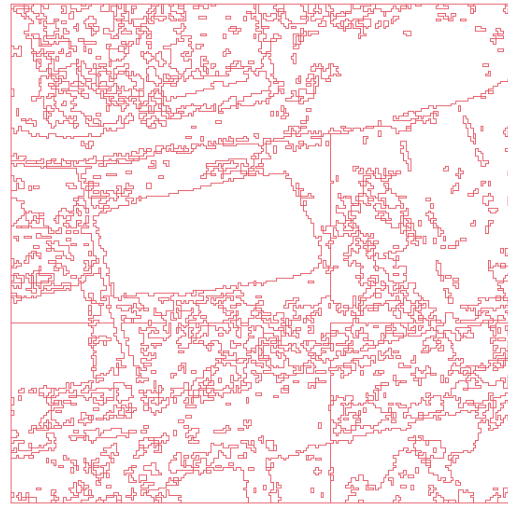
(b) 90fv8
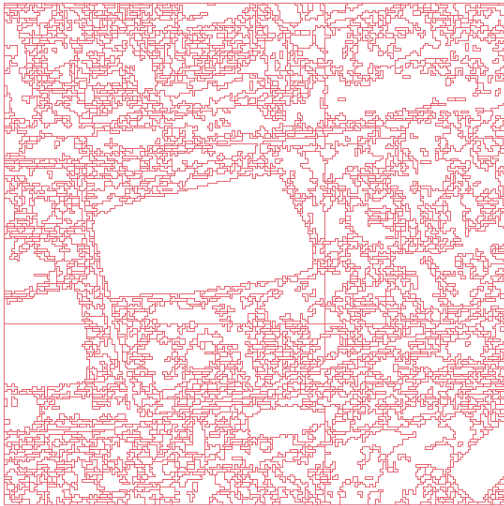
(c) 30mv4

(d) 30mv4-5
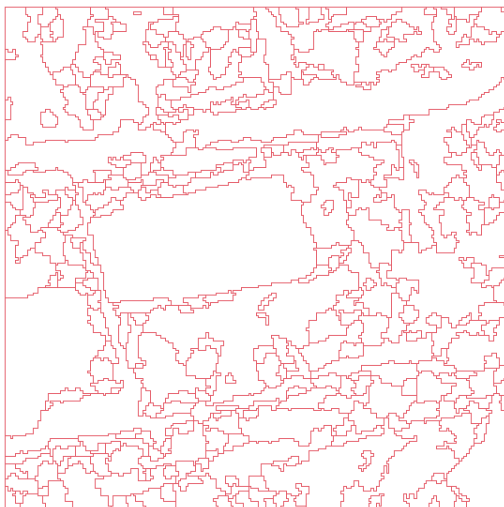
(e) 30mv4-10
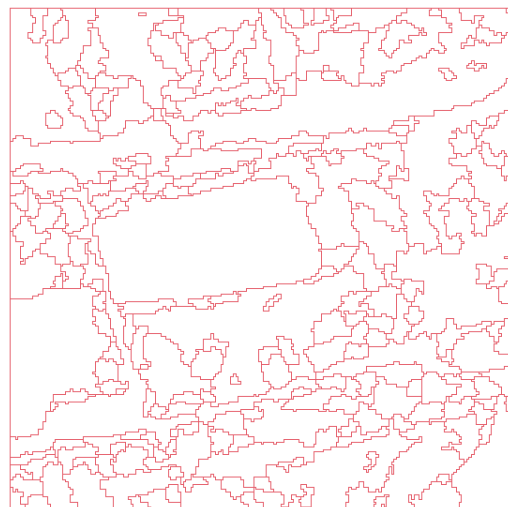
(f) 30mv4-15

Figure 4.7 - SegmenTHIS segmentations of IKONOS image using various parameters.

To compare the changes of the new features of SegmenTHIS, Figure 4.8(a) was segmented using original option (called here "1M4") and some other options ("1F4", "3M4" and "3M8"), with an instance of each respectively shown in Figures 4.9(a), 4.9(b), 4.9(b) and 4.9(c). The test images was repeated 5 times and each instance was combined with each other. Results shown in Figure 4.10 present the best performance for the last four cases (i.e., the randomness testing of the same image). The labeled "F-M" is the comparison between the instances of "1F4" against "1M4" segmentations. Visually the "1F4" achieved the best segmentation because it is possible to identify the main building. Souto's "1M4" case made confusion with background with the segmentation threshold of $L = 25$ used in all tests. The graphic (Figure 4.10) also shows that there is a significant change in the indexes of the SegmenTHIS enhancements, compared to the randomness case (the codes with the "r" appended).

(a)   (b)
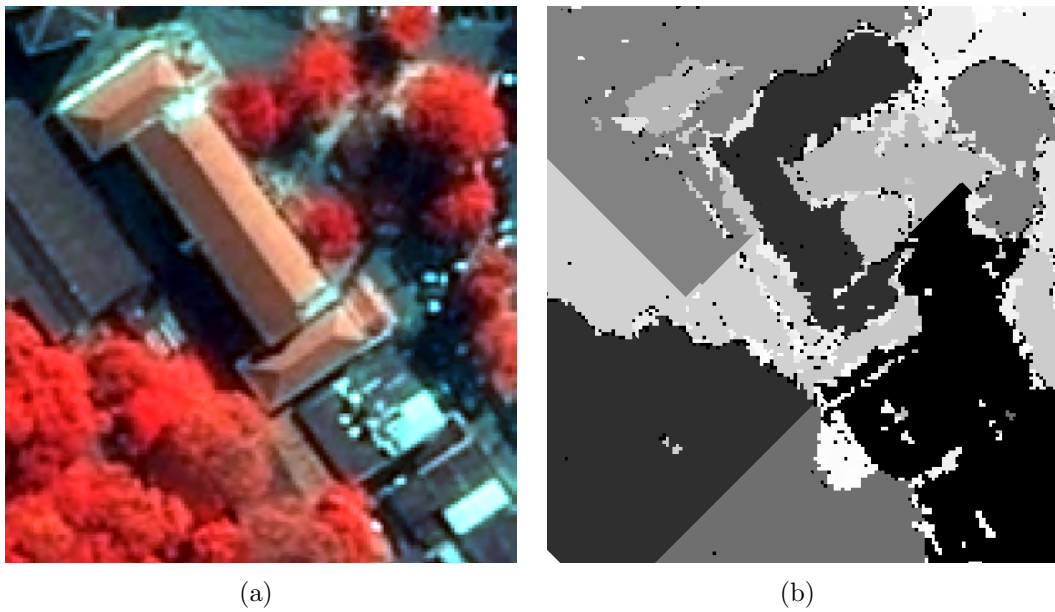
Figure 4.8 - Test image and original hue segmentation: (a) image; and (b) Souto (1M4).
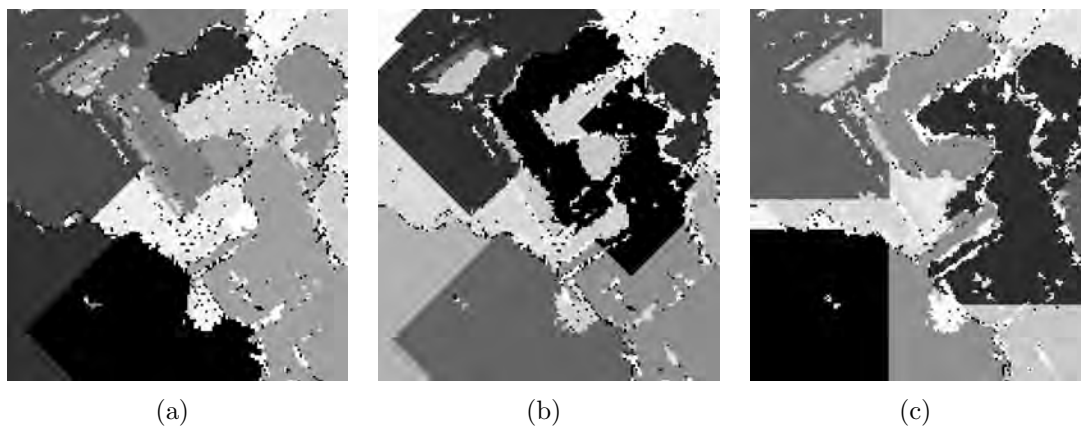
Figure 4.9 - Results: (a) THIS 1F4; and (b) THIS 3M4; and (c) THIS 3M8.
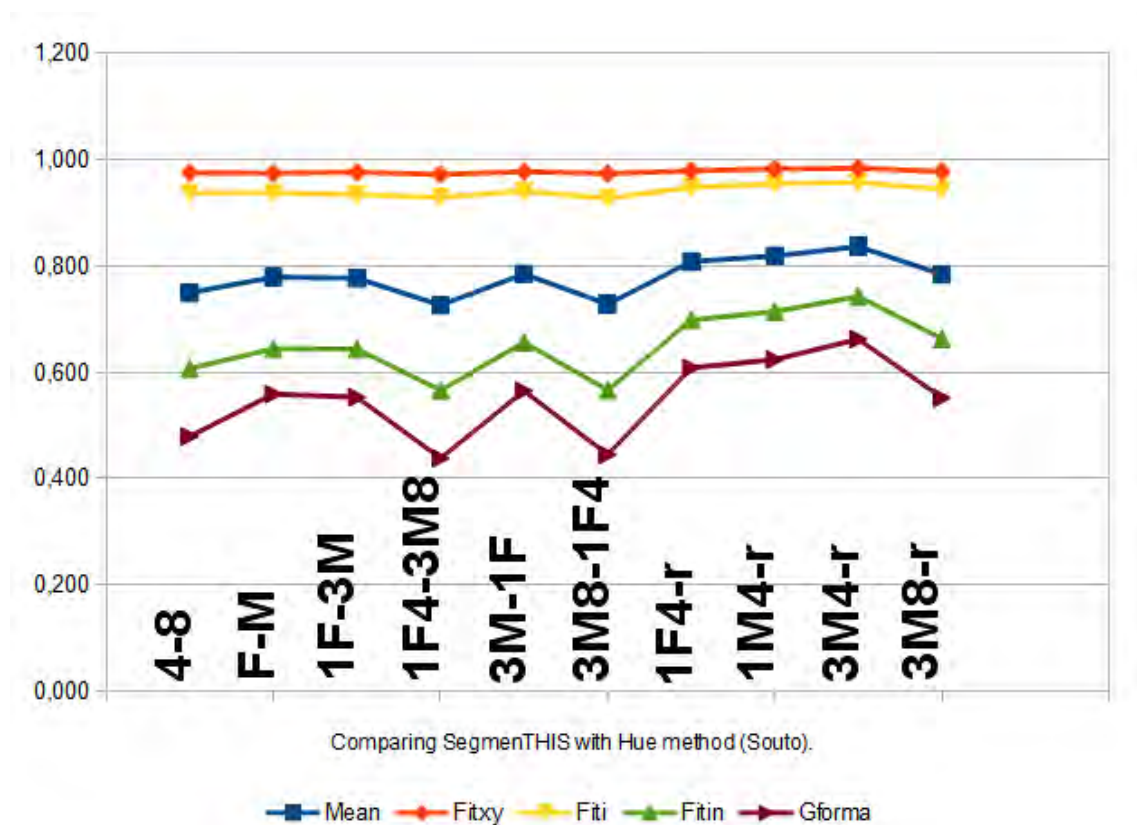
Figure 4.10 - Results of SegmenTHIS and hue segmentation (Souto).

4.2 Evaluation of the techniques for pre-processing of multispectral images

(a) Hue image   (b) Hue after image normalization
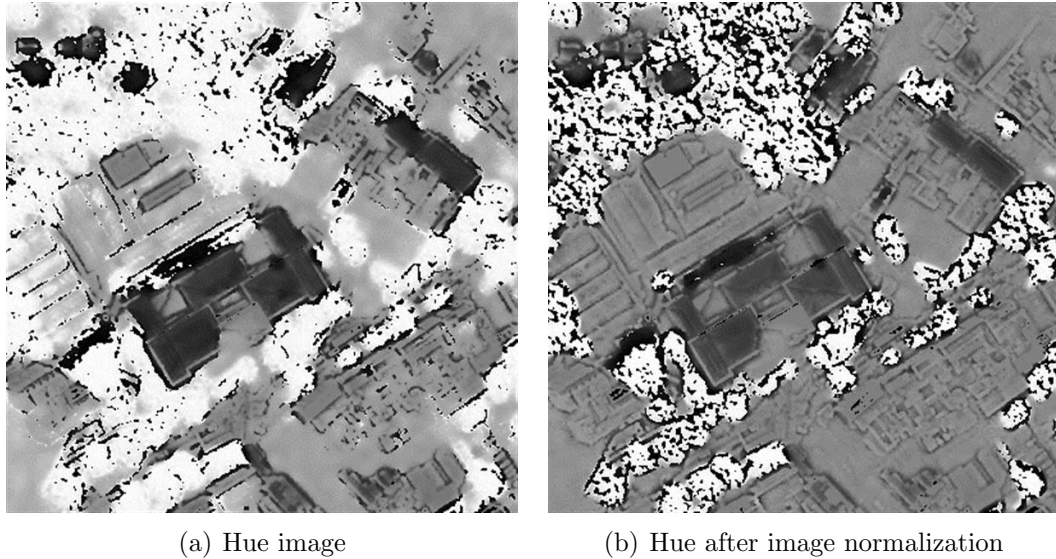
Figure 4.11 - Hue channel obtained from composition 431 of multispectral bands.

Figure 4.11 shows a hue image and a hue image after image normalization. It has been observed that the results are quite different after a normalization and frequently the normalized image gives better results. However in some cases the normalization produces a worse segmentation. The case shown in Figure 4.11(b) have the vegetation emphasized but the asphalt mingles more with the roofs.

Figure 4.12 present a zoom of the same building without and with previous pre-processing normalization on multispectral bands. The change in the hue is easily noticeable and the product becomes quite different.

Synthetic hue of Figure 4.13(a) highlights more variety of targets than the hue shown in Figure 4.11. Comparing segmentation shown in Figure 4.13 with the one presented previously in Figure 4.3, the result is less fragmented, but the merge feature made larger but more noisy segments.

Figures 4.14(a), 4.14(b), 4.14(c), 4.14(d) are the principal components of the selected scene, and the first channel is presented in Figure 4.14(a), and have, by definition, greatest variation. Notice that the figures presented here are resampled for intensity data to supply visual enhancement, making it possible to visualize spatially the darker and the lighter areas. Only having this in mind it is possible to compare

90
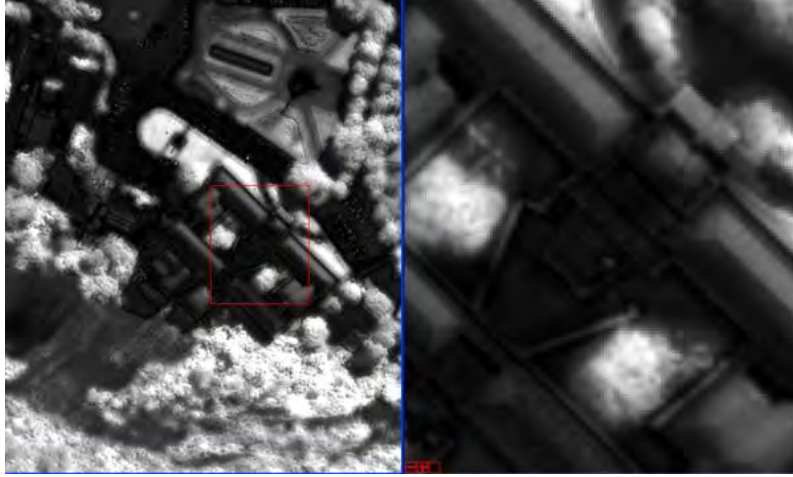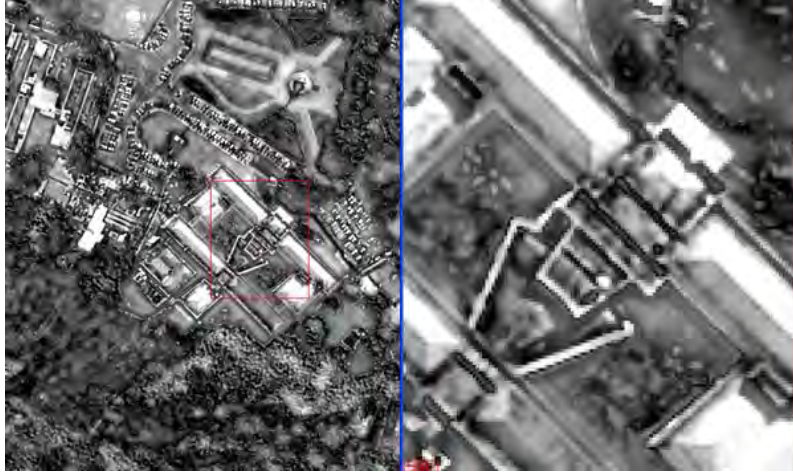
Figure 4.12 - Synthetic band created: (a) based on unchanged bands; (b) based on normalized bands.

the following figures. While in Figure 4.14(a) the vegetation appears sharp, in Figure 4.14(b) vegetation appears blurry. The shaddow effect in the building (see the right image) is evident only on the first component (Figure 4.14(a)), but there is only a subtle effect of shadow in all other images (Figures 4.14(b), 4.14(c), 4.14(d)). Figures 4.14(a) and 4.14(a) presents a complementary aspect, one resembling the negative of the other. This is common to happen with the first two principal components. It can be observed for some objects, like the sidewalk of the plaza and the white spot on the parking cars, where high values in one correspond to low values in the other and vice versa.

Most images presented in this thesis are stretched to the byte interval $\{0 \cdots 255\}$, but

(a)   (b)
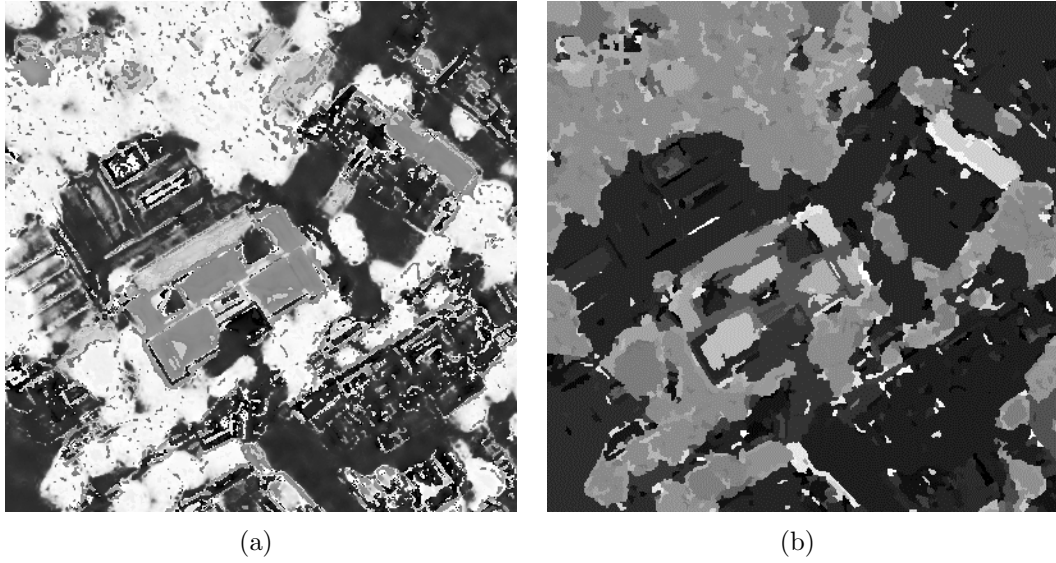
Figure 4.13 - (a) S-Weighted hue synthetic channel; and (b) S-Weighted segmentation in SegmenTHIS with $p = (L20P10)$.

Figures 4.14 the image become a bit more noisy. The vegetation is very deteriorated. It is not possible to use this channel to get details about the vegetation, but maybe it is possible to find a mask on vegetation using second component (Figure 4.14(b)).

Figures 4.14(d) and 4.14(e) are the last two principal components and hence most impure. Still it is possible to identify several features, so this means the fourth band is contributing with valuable data. If it was a linear combination of other three the principal component would have eliminated any relevant information from the last principal component. So this can justify the use of one more band with advantage. Figure 4.14(f) is a color composition using 1st, 2nd and 4th channels of the principal components.

Table 4.1 is associated with Figure 4.15. The first principal component brings with it 66% of the entire information present in the multispectral set of four images. The second adds 32% to the overall information. The third adds only 1%, meaning the high correlation existing. After stretching the histogram to display the images shown in PC3 (Figure 4.14(c)) and PC4 (Figure 4.14(d)), the geometric aspect of the image appears, but is very noisy and adds very little new information. The fourth have practically no new information (0.10%).

(a) First principal component.

(b) Second principal component.

(c) Third principal component.

(d) Fourth principal component.

(e) Principal components 132.

(f) Principal components 432.
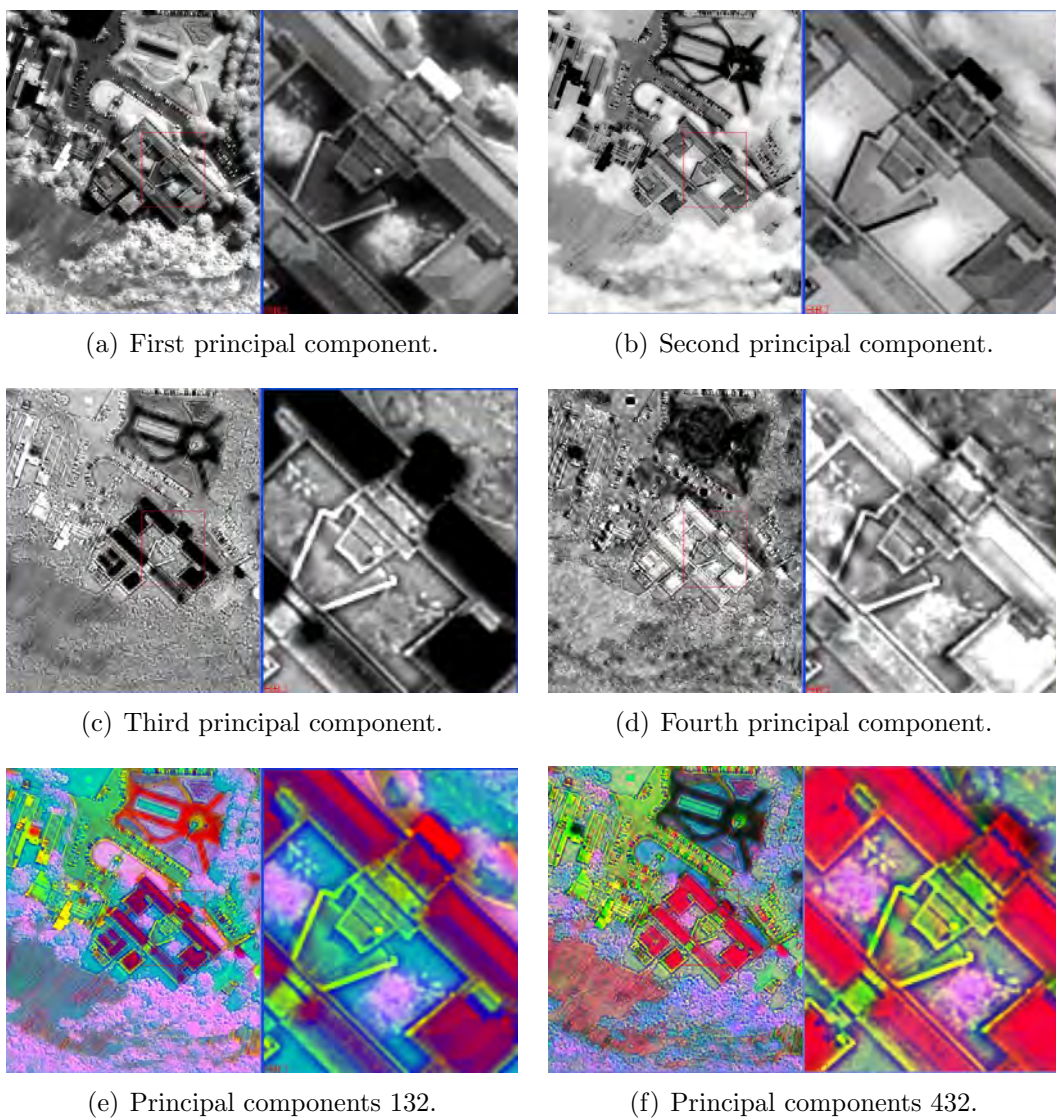
Figure 4.14 - PCA analysis channels.

Table 4.1 - Percent of output PC channels

PC	Eigenvalue	Percent	Overall
1	137013	66.67%	66.67%
2	66269	32.24%	98.91%
3	2044	0.99%	99.90%
4	196	0.10%	100.00%

4.3 Evaluation of the IHS segmentation algorithm based on S-Weighted and Sine-Cosine methods
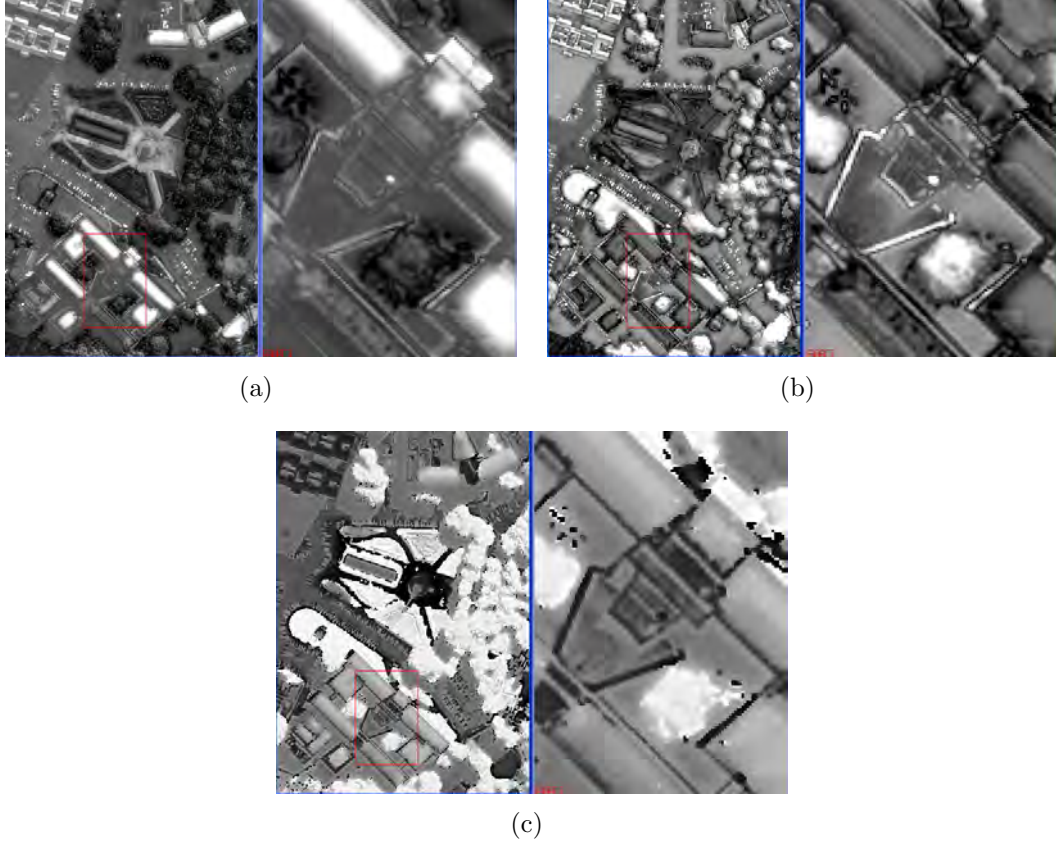
Figure 4.15 - Gray-level images of: (a) intensity (value) channel on IHS space; (b) saturation channel on IHS space; and (c) hue channel on IHS space.

Figure 4.15 shows the three channels of the transformation RGB–IHS of the image input (Figure 4.1). By visual verification the intensity channel in some areas is correlated with both intensity and hue images, and could be used in future works as the base of segmentation (i.e., instead of hue). This relation and coherence in saturation image is the reason why the S-Weighted method was proposed to provide an alternative input to the hue channel segmentation method.

Figures 4.17(a), 4.17(b) 4.16, 4.18(a), and 4.18(b) shows the following channels: Intensity, Saturation, Hue, Sine, Cosine, Hue.

Images in Figure 4.17 carries all the information needed to reconstruct the hue image. The equation $f_{sin}^2 + f_{cos}^2 = 1$ relates sine with cosine and $hue = \arctan(sin/cos)$.
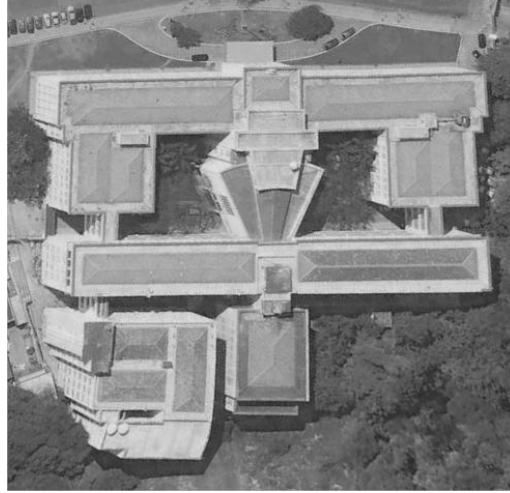
Figure 4.16 - Intensity image.

(a)   (b)
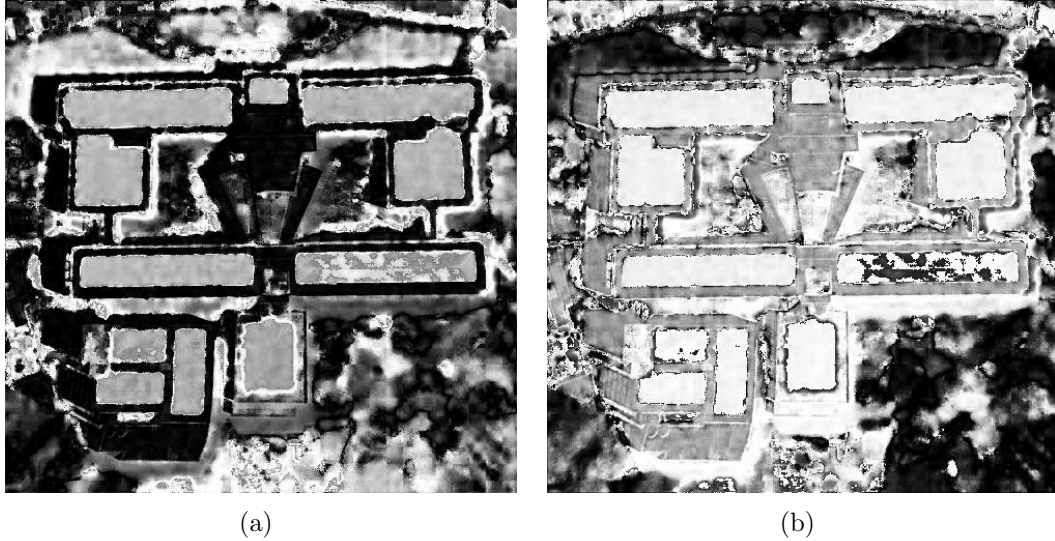
Figure 4.17 - Sine and cosine images stretched to byte format interval: (a) sine image; and (b) cosine image.

S-Weighted method is an alternative way to obtain the hue channel from more than 3 multispectral bands. Moik (1980) extended the classic IHS equations to add new bands, redefining the angle, but we proposed a saturation-weighted mean hue based on hues obtained by the combination of all 3-combination of a set of the n available multispectral bands.

Figure 4.19 is the synthetic hue image obtained by S-Weighted method, and it can be compared with other hue channels.

95

<table>
<tr><td>(a)</td><td>(b)</td></tr>
</table>
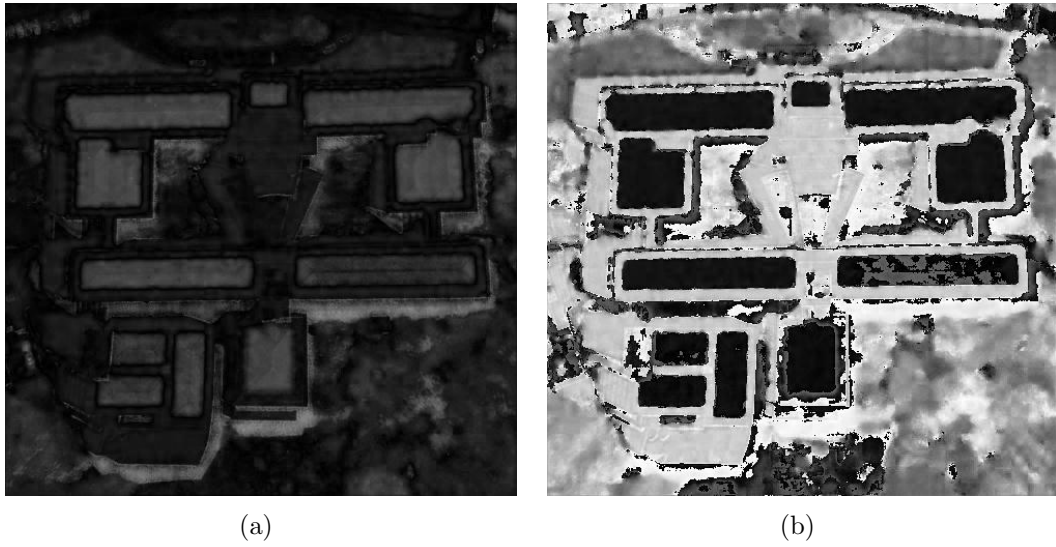
Figure 4.18 - IHS transformed images: (a) saturation; and (b) hue.
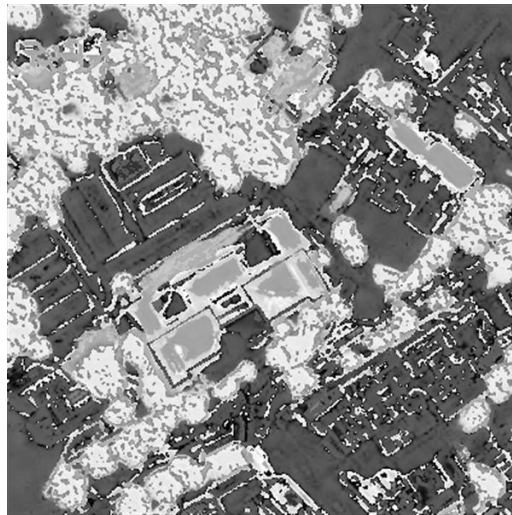
Figure 4.19 - Synthetic image obtained from multiple hue channels.

Segmentations produced by Moik method and Sin-Cos method in Figure 4.20 are based on hue and SPRING region-growing segmentation is based directly on all the 4 multispectral bands. Based on visual examination of the three images, where we seek parameters that roughly equalize the results in terms of the degree of land fragmentation, it is concluded that:

a) vegetation is more homogeneous in both segmentations based on hue (a) and (c);
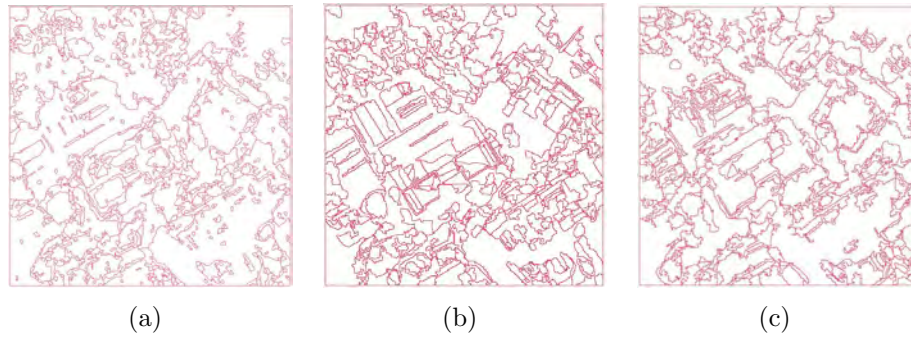
Figure 4.20 - Comparing segmentations: (a) Moik method; (b) SPRING segmentation; and (c) Sin-Cos method.

b) urban delimitation of roofs is more precise in SPRING (b) segmentation;

c) some areas of Sin-Cos (c) is over-segmented than the others;

d) Moik method (a) contains many small fragmented areas.

Figures in this section (Figures 4.21 and 4.22(b) are results of Spring and Segmen-THIS to show results of Sin-Cos method and Hue S-Weighted methods.
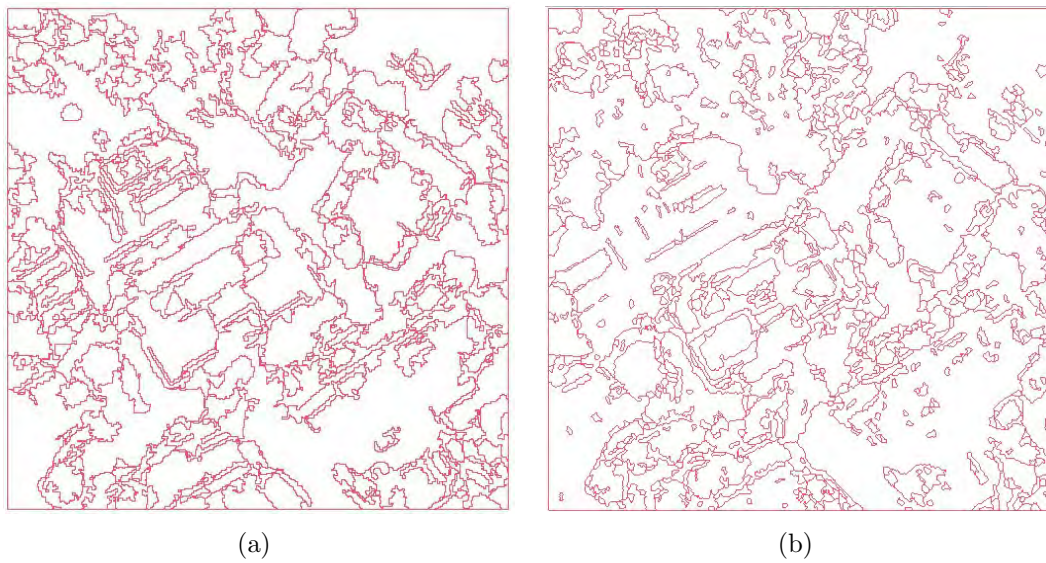
Figure 4.21 - Spring and SegmenTHIS used to compare Cos-Sin and Synthetic Hue (S-Weighted): (a) Cos-Sin using Spring; and (b) S-Weighted using SegmenTHIS.

(a)   (b)
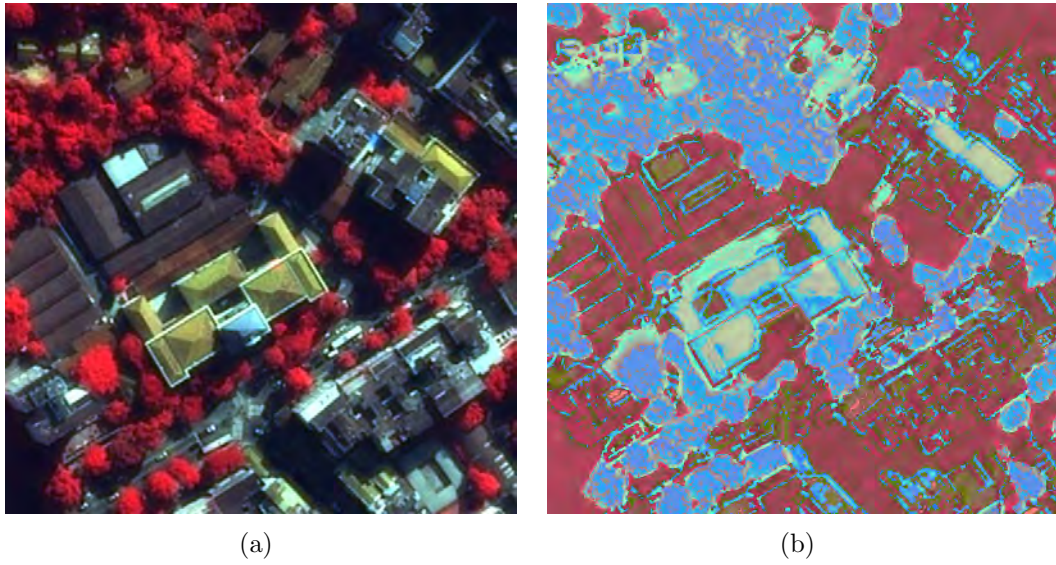
Figure 4.22 - Image and synthetic image: (a) Plèiades RGB composition 431; and (b) Synthetic Hue (S-Weighted) in colors.

For visual comparison, Figure 4.22 shows the composite 431 placed beside the S-Weighted synthetic hue. This picture is displayed using a randomly chosen color table to allow identification of the prevailing features present in the sample, where vegetation becomes bluish and reddened the asphalt.

(a) Cosine image   (b) Sine image
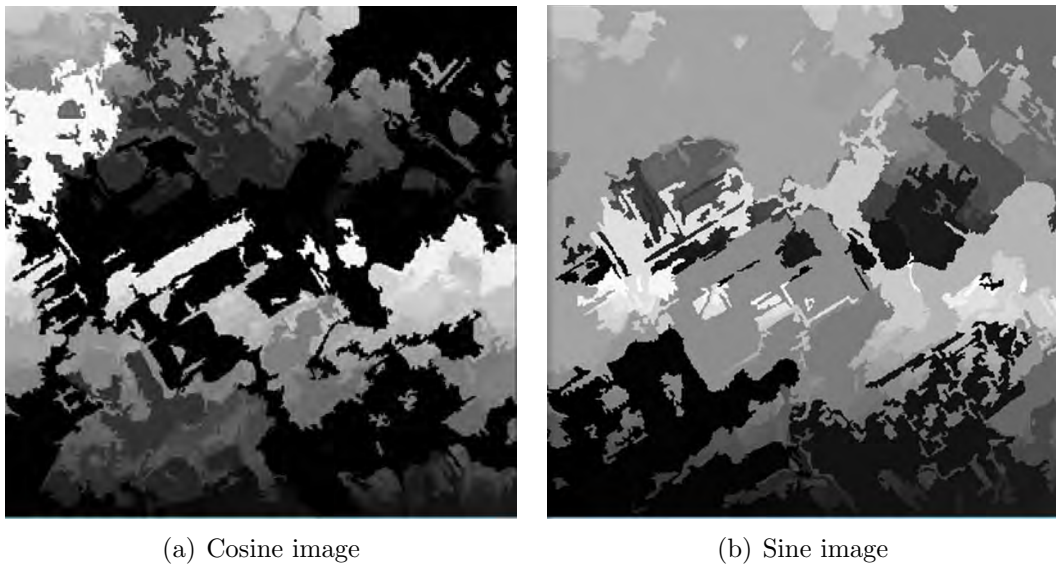
Figure 4.23 - Segments of the (a) cosine and (b) sine of the hue.

(a) With a gray colortable "A"   (b) With a gray colortable "B"
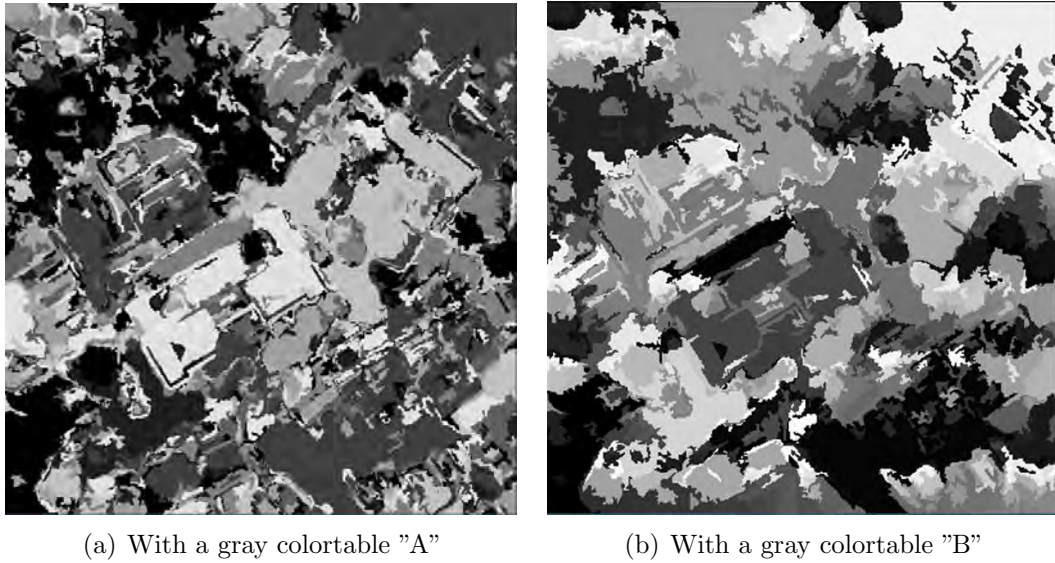
Figure 4.24 - Segmented image created mathematically by the intersection of the results of sine and cosine segmentations.

All examples was tested using SegmenTHIS, SPRING and ENVI softwares. Cosine and Sine images are shown in Figure 4.23 in a gray gradient color table, with a color-table where darker regions correspond to segments with lower ID. The Sine and Cosine images was segmented and merged (Figure 4.24) to simulate a hue-based segmentation using any segmentation algorithm 1-band capable. The same segmentation result is displayed with different color tables in Figure 4.24(a) and Figure 4.24(b).
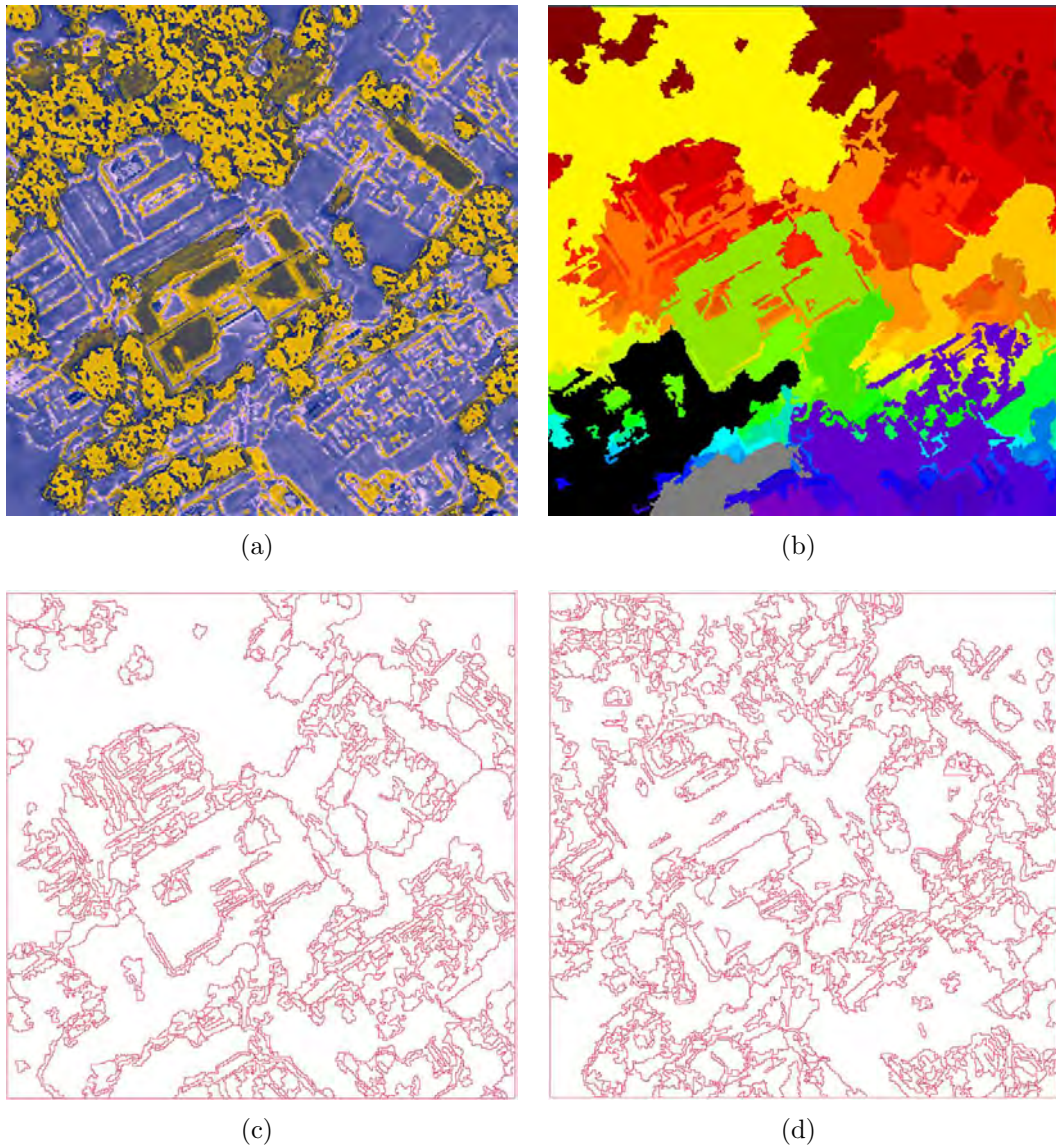
Figure 4.25 - (a) Synthetic hue channel in color; (b) segmentation output associated with
a randomly colored color table; (c) SPRING segmentation of sine channel
(contours); (b) SPRING segmentation of cosine channel (contours).

Figures 4.23, 4.24 and 4.25 shows segmentation and intermediate products of the
segmentation using Sin-Cos method and Synthetic Hue S-Weighted Method for com-
parisons. The results are shown in Figure 4.26.
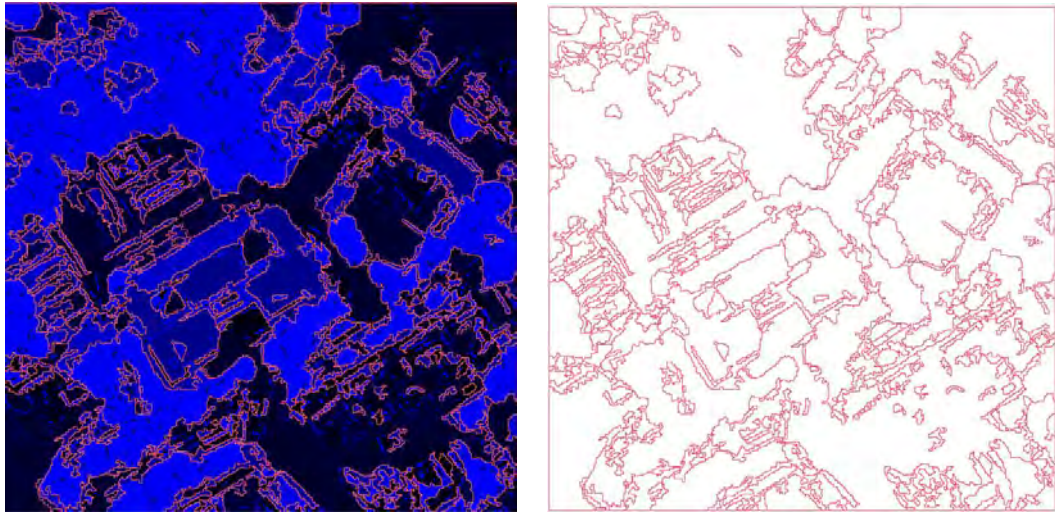
(a) Segmentation contour and blob of Spring segmentation of hue

(b) Segmentation contour visualization of Spring segmentation of hue

Figure 4.26 - Segmentation outputs.
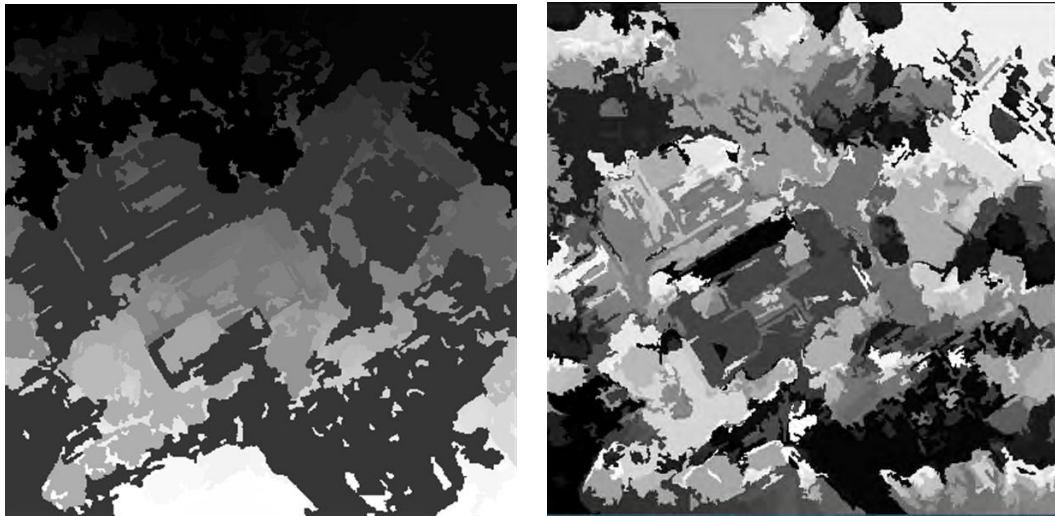
(a)

(b)

Figure 4.27 - Results: (a) Hue segmentation in SegmenTHIS. (b) Sin-Cos segmentation.

The comparison between the products would become a more precise task if the segmentations was more similar, however the sensitivity to a large number of parameters and lack of of a defined criterion of comparison makes it difficult to compare the segmentations. Thus, the visual comparison can be differentiated by details and examples and describe the general appearance of the results.

4.4 Quantitative analysis of the segmentations

Although there are other quantitative methods to measure segmentations, there is no consensus about the effectiveness and superiority of any single method to evaluate segmentations.

Quantitative assessment have limitations because there are several reasons that may affect the results. The present method based on $Fitxy$, $Fiti$, $Fitin$ and $Gforma$ indexes was readily available for the tests and requires only the existence of three input images: the original image, the reference image and the segmented image.

The implementation used shown itself to be very stable and it is a quantitative way to compare segmentations indeed. However, it is noticeable the high variance of the segmentations, indicating the need to evaluate each case in order to define statistical criteria of quality.

To assess segmentation accuracy the method using the indexes Fitxy, Fiti, Fitn and Gforma (DELVES et al., 1992) an IDL implementation was used, with the output of the mean and standard deviation for each case.

To evaluate quantitatively the changes in segmentation by these indexes, several distinct segmentations was submitted to compute the metrics and assess the segmentation overall accuracy, each one to find the best performance. The results are presented in blocks of segmentations by type of comparison and reference file used.

For each comparison the values of the indexes of all segments of both reference file and test files is computed and the mean value of the four indexes ($Fitxy, Fiti, Fitin and Gforma$) is computed and summarized into a general mean (referred on the graphics as $Mean$ index), the arithmetic mean of the four indexes.

These values correspond to the performance of the given segmentation based on the reference and so are dependent on the quality of the reference file. The reference file is "the truth" we want to get back from the segmentation software, so there are unlimited possibilities.

We defined segmentation simulations by delimiting the visually detected regions by creating a new manually segmented image. By this, the operator experience is being considered and the new created file used as reference for the tests. This approach allows the comparison based on the user own performance. We tested the manually defined reference to find out how close the segmentation algorithm is and for each file

compared to the reference we obtained a segmentation performance set of indexes and the general mean.

The process considers every pair of combination of for each parameter to find the best fitted combination. Using the $Gforma$ index a matrix with $m\dot{n}$ elements is builded, where m is the number of segments of the reference and n is the number of segments of the segmentation. This matrix contains all possible combination, that is used to find the best match of the m with the n segments, and for this a new matrix Fit is created. More details are described with examples by Lucca (1998).

Finding the segments of the segmentation image related to the reference segments, considering only the segments of the reference file (i.e., r_i), the values of each $Fitxy(r_i, m_j)$, $Fiti(r_i, m_j)$, $Fitin(r_i, m_j)$ and $Gforma(r_i, m_j)$ are used to find the final $Fitxy(r_i, m_j)$, $Fiti(r_i, m_j)$, $Fitin(r_i, m_j)$ and $Gforma(r_i, m_j)$, for all i segments of the reference image and for the corresponding j defining the corresponding m_j. The final $Fitxy$, $Fiti$, $Fitin$ and $Gforma$ is the mean of these values, computed with the variance.

The use of metrics that give the same value to regions of distinguished significance on the purpose of segmentation may be a biased choice.

If the result of the segmentation is adequate in the main area of interest, but it is bad in other image regions (e.g. in most of the image), then the metrics would indicate a poor outcome, despite the eventual superiority of the method whether the objectives were taken into account as well..for the objective.

The use of metrics weighted according to the importance of visual information based on context (CAVALLARO et al., 2002) could be a solution.

Figure 4.28(a) shows in blue the mean index for all parameters of segmentations produced by SegmenTHIS prototype; in red the results of the index $Fitxy$; in yellow the results of the index $Fiti$; in green the results of the index $Fitin$; and in brown the results of the index $Gforma$. The colors represent the metrics, and the continuity of the line shows the upper and lower bounds for one standard deviation above and below the mean. The continuous line represent the upper and the dashed line indicate the lower bound where is only 68% of the values.

The continuous and dashed lines was supposed to be much closer for a segmentation to make conclusions about the accuracy of the product.

Figure 4.28(a) is the result of testing a manually defined reference with several segmentations results and the best suited is the case $l0p10$, which means thresholding of 10 degrees on hue and a merge parameter for every segment with less than 10 pixels. A sketch of the contour analyzed is shown in Figures 4.28(b) to 4.28(e).

Figure 4.29 is the result of SPRING segmentations, indicating the best parameters configuration with Fitin better than 70%. This was achieved by the test using the synthetic gaussian images (Figure 4.30), a result almost linear, with no significant difference among the tests.

Figure 4.31 is the numeric result of the test of 82 distinct parameters of SegmenTHIS, to fine tuning the parameters. The best cases are a bit worse than SPRING results and Figure 4.32 for the other test of the manual segmented reference was inconclusive because there is no acceptable set of parameters. This case indicated that most of the segments are not well fitted, even if some are very good. A new metric must be used to find a better way to compare the parameters for these cases.
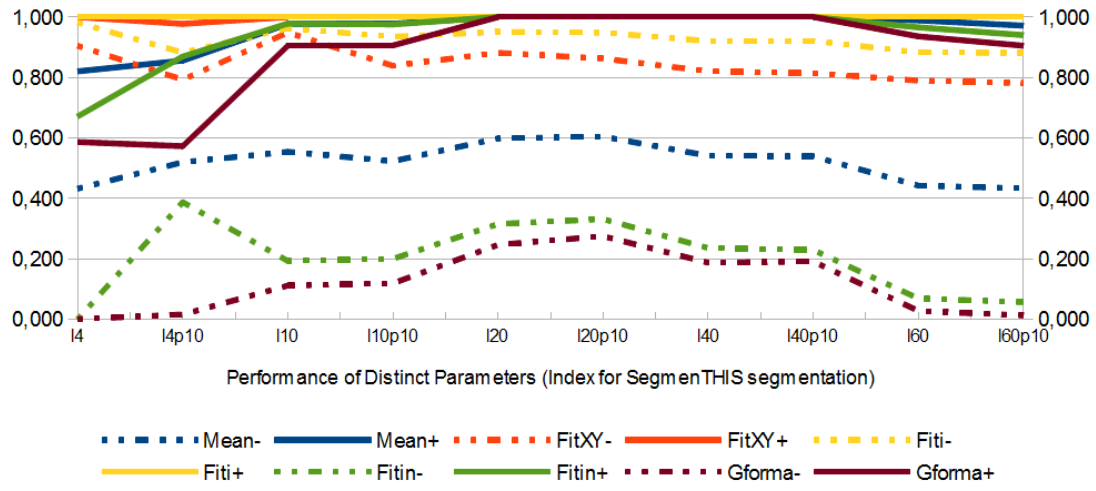
(a)
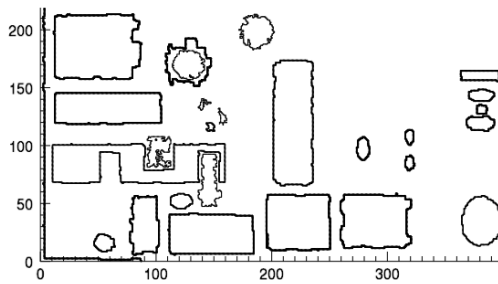
(b)
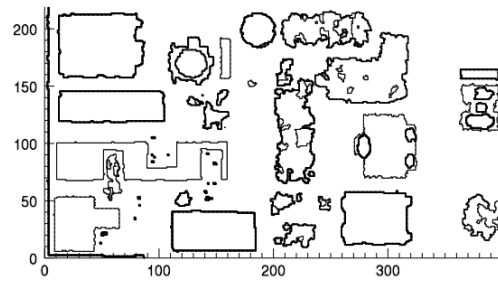
(c)

(d)
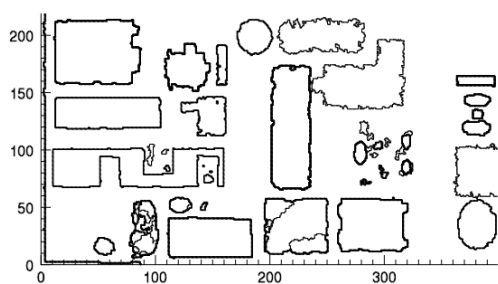
(e)

Figure 4.28 - (a) Testing results of SegmenTHIS parameters of thresholding and merging, using a manually defined reference; (b–e) examples of contours of segments used to compare to a reference: (b) segment with $L40P10$; (c) segment with $L10P10$; (d) segment with $L20$; (e) segment with $L20P10$.
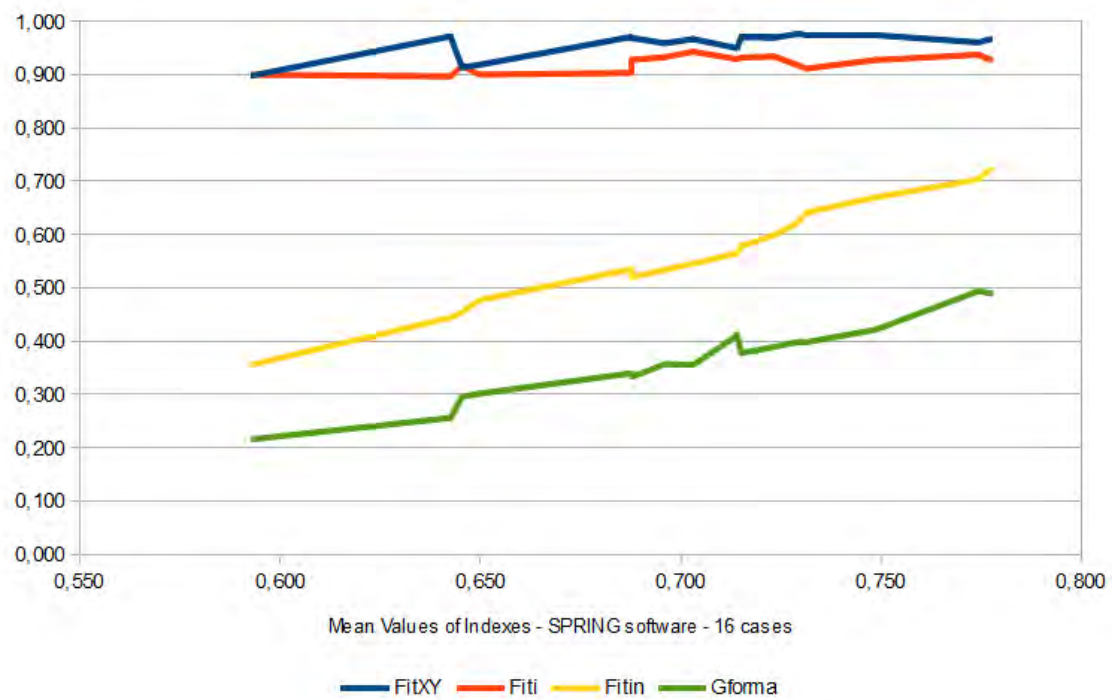
Figure 4.29 - Result of FitXY, Fiti, Fitin and Gforma for Spring segmentation.
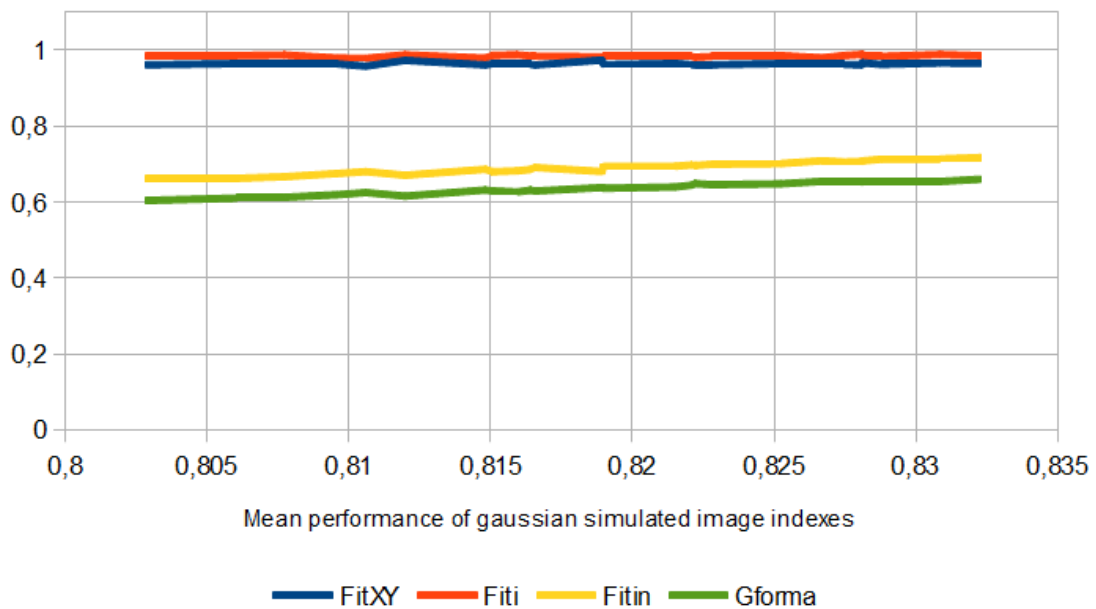
Figure 4.30 - Results of FitXY, Fiti, Fitin and Gforma for 25 sintetic images based on the same phantom and ROIs over an IKONOS image.
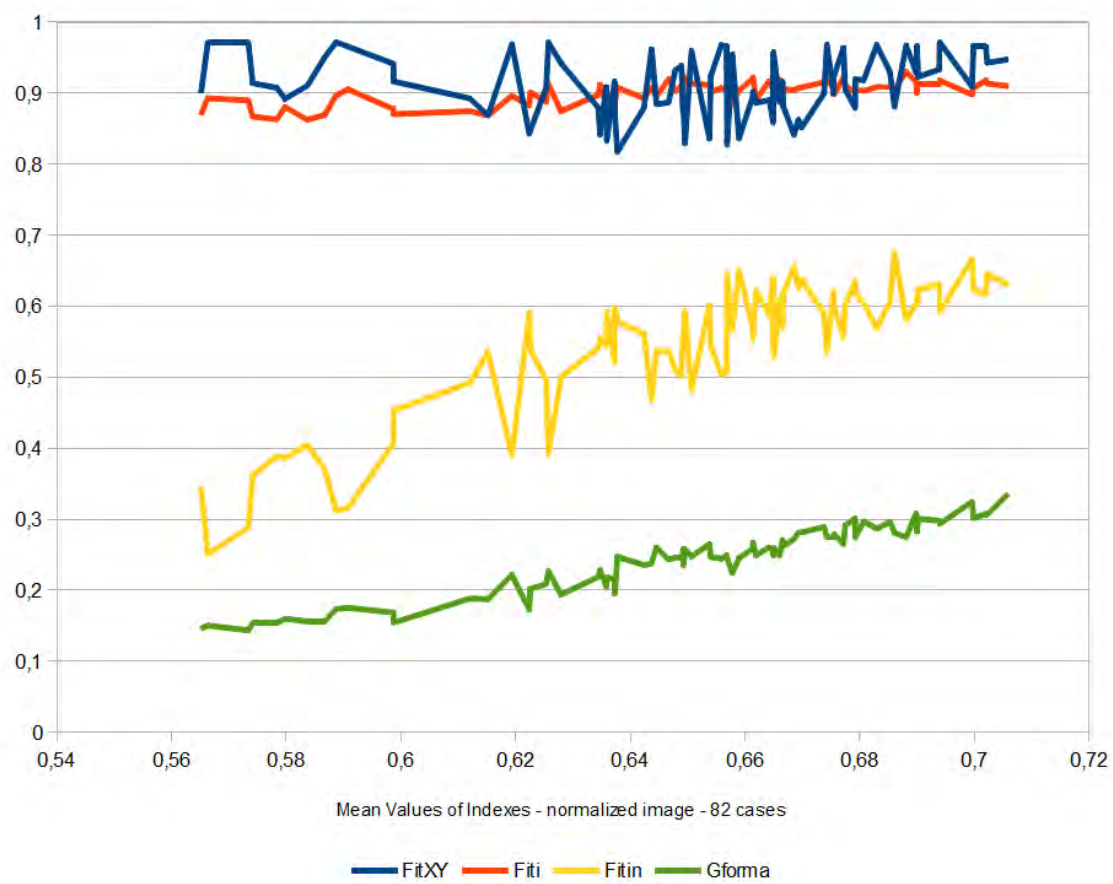
Figure 4.31 - Testing 82 distinct combinations of parameters.
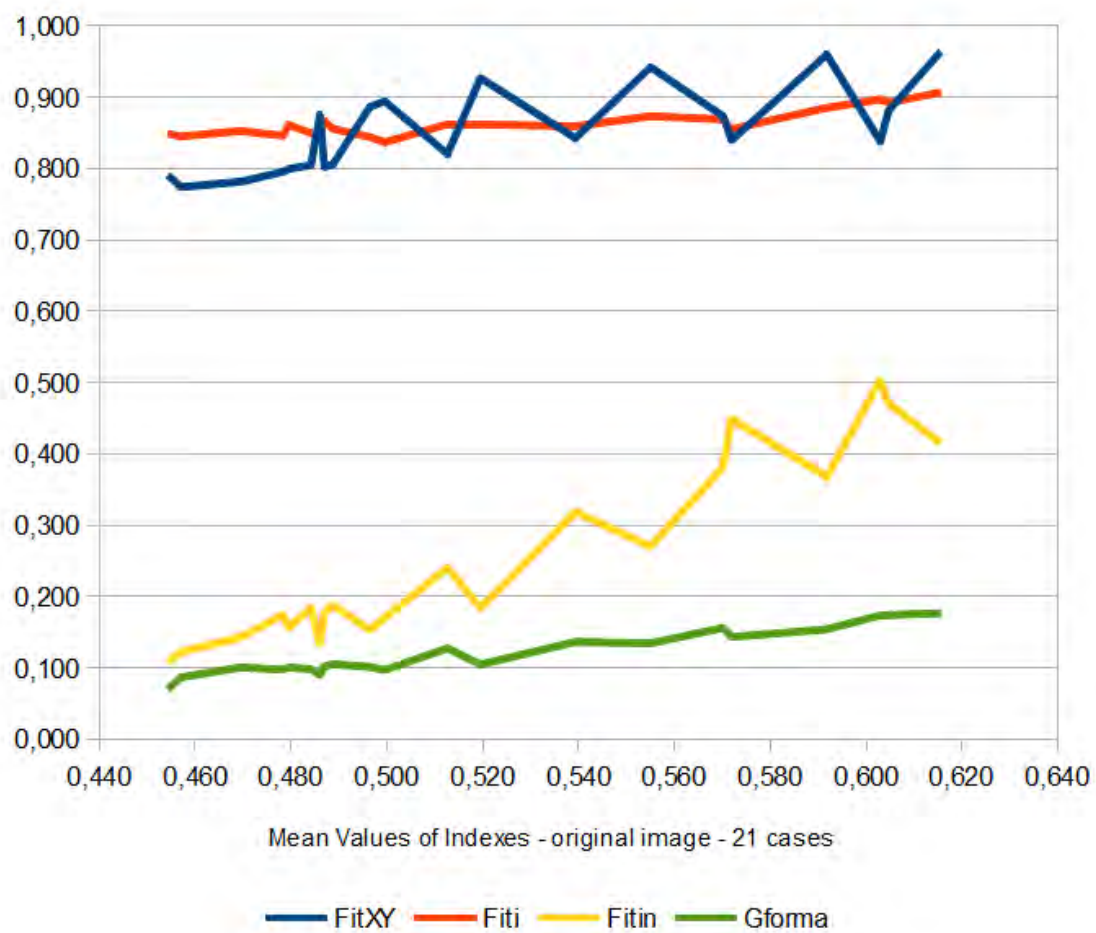
Figure 4.32 - Testing 21 combinations of parameters without previous image normalization.

4.5 Turning Function results

Figure 4.33 presents a noisy case of a rectangle (VOLOTAO et al., 2010a). This picture represent a synthetic noisy shape to demonstrate the patterns of each part of the figure. A, B, C and D are the sides of the original rectangle (lower left). Most of the perimeter of this shape corresponds to noises.

Figure 4.34 shows an example of a rectangle with strong noises effect with the letters $\{A, B, C, D, E, F, G\}$ to represent it. A concavity in the contour is represented by a descending stair in the turning function. Circles or arcs are mapped as ascending and aligned steps Spikes do greatly increase the perimeter and are easily identifiable by the analysis of the turning function. Distinct kind of noises may have distinct characteristics but each of them can be found by search on turning function.
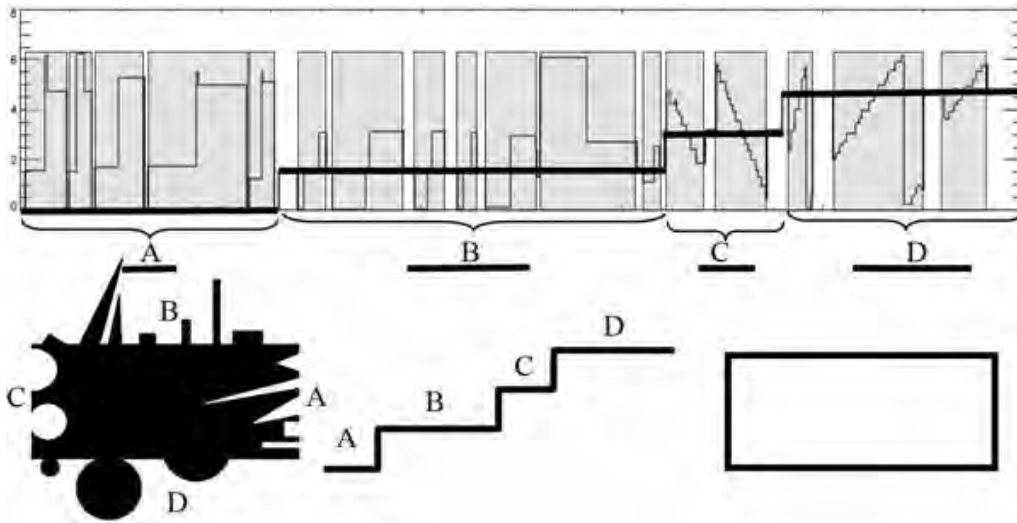
Figure 4.33 - Rectangular polygon representing a shaped contour with 17 noises.
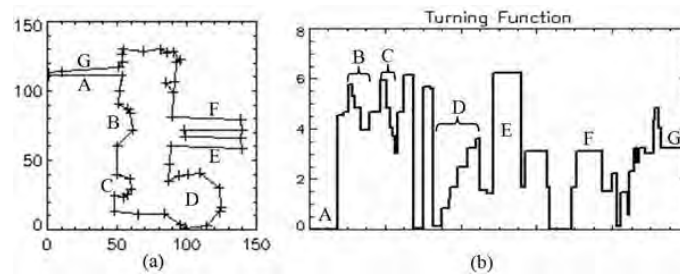
Figure 4.34 - (a) Contour representation of a rectangular object with noises; (b) corresponding turning function. Regions A, B, C, D, E, F and G are marked in both cases.
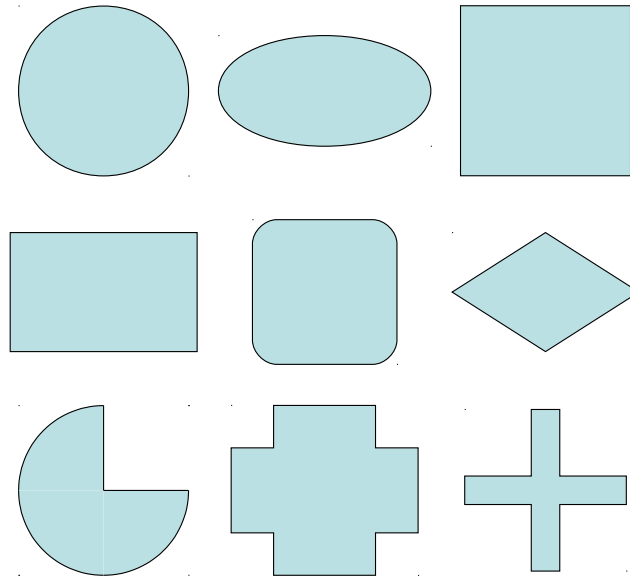
Figure 4.35 - Image with some shapes used to compute and compare indexes and shapes.

Figure 4.35 is an example image used to extract indexes to make shape analysis. Each object produces its own shape parameters and this allows the comparison and categorization of the shapes. The template was segmented using SegmenTHIS setting the option "blob", going to a supervised mode to grow the chosen region. The indexes developed are important to provide a way to optimize the shape in the segmentation process. Little changes can be made for test whether the index become more likely to the desired shape category.

Different objects extracted from Figure 4.35 were segmented to produce noisy shapes. These noisy shapes enable the study of the shapes and patterns for recognition and refinement purposes. Figure 4.36 is a circle, Figure 4.37 is an ellipse, Figure 4.38 is a square, Figure 4.39 is a rectangle, Figure 4.40 is a square with round borders, Figure 4.41 is a diamond, Figure 4.42 is a sector, Figure 4.43 is a large cross, Figure 4.44 is a cross

All figures have noises. Using the right figures it is possible to identify the irregularities. All regular shapes are well behaved on its turning function. Note that it is easier to find a noise using a turning function than seeing the corresponding shape. This property is what this thesis explores. Note the gray little square on each figure: it is the starting point of the turning function.

(a) Blob   (b) Turning function
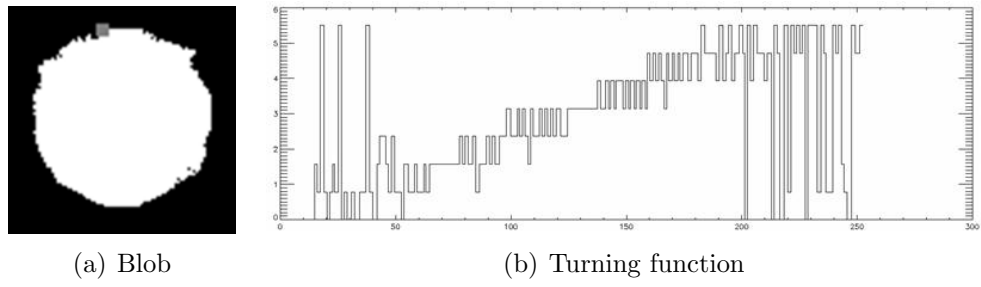
Figure 4.36 - Circle with noises: a circular shape example.

Figure 4.36 exemplifies the new geometry presented in a plot for a circular-shaped pattern. There is a clue to indicate that the representation was obtained by the 8-angles chain code algorithm and then converted into the turning function representation: that there are exactly eight equidistant steps, including zero. This is the same condition to revert the polygon to a chain code again. In this angle-limited representation the curves and sloped lines exhibit an oscillation of angles around the fixed values that are more approximate, as can be seen in Figures 4.36 to 4.44(b).

(a) Blob   (b) Turning function

Figure 4.37 - An ellipse.

(a) Blob   (b) Turning function
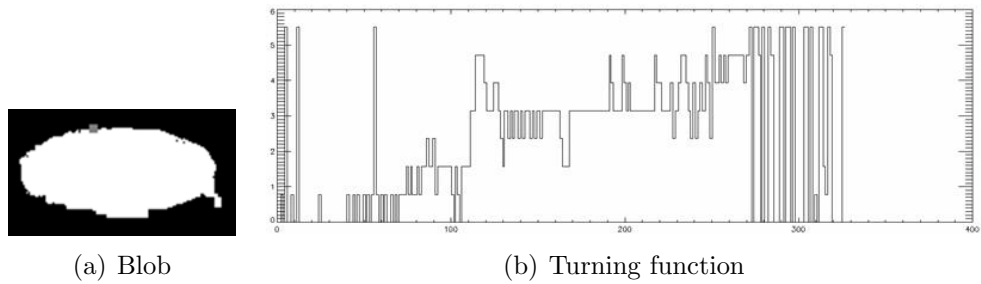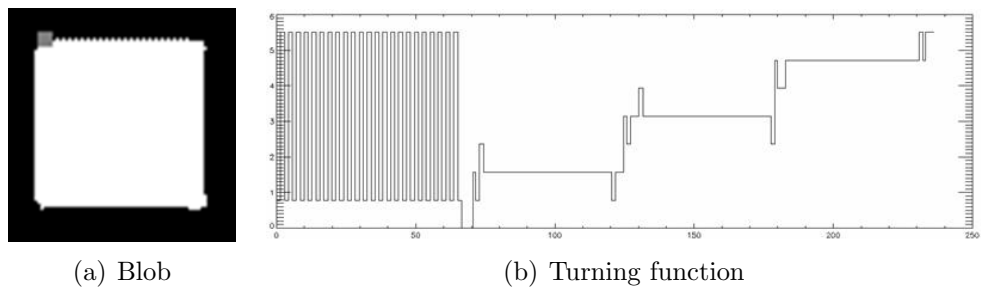
Figure 4.38 - A blob with square shape.

111

(a) Blob   (b) Turning function
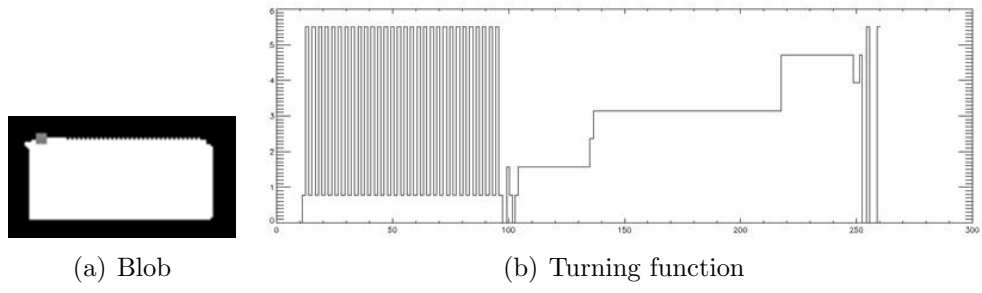
Figure 4.39 - A rectangle shaped blob.

(a) Blob   (b) Turning function
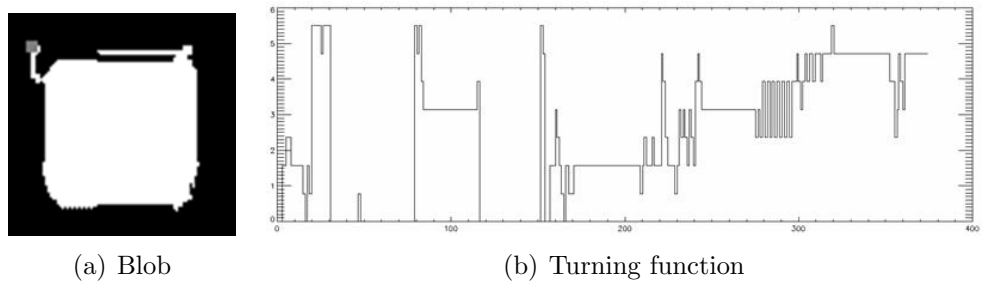
Figure 4.40 - A rounded square shaped blob.

(a) Blob   (b) Turning function
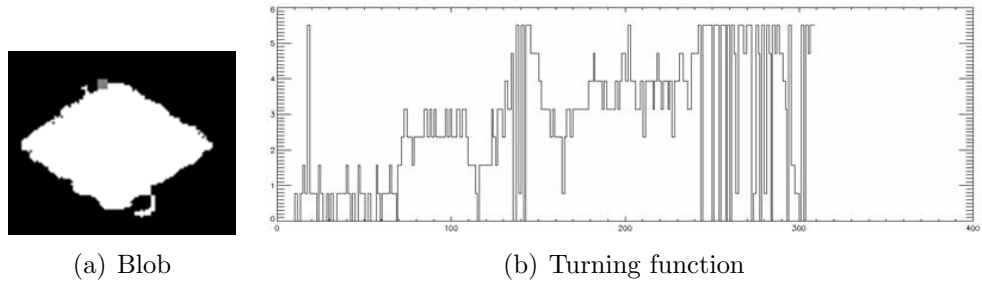
Figure 4.41 - A rhombus shaped blob.

Tables 4.2, 4.3, 4.4 and 4.5 are related to the Figures 4.36, 4.37, 4.38, 4.39, 4.40, 4.41, 4.42, 4.43, and 4.44.

(a) Blob
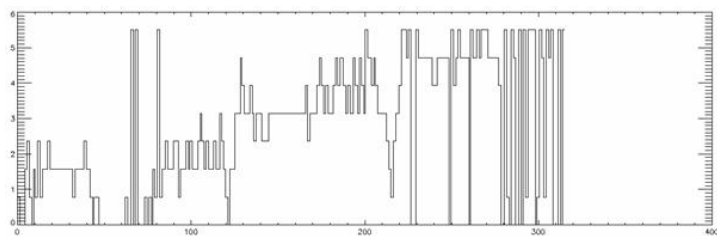
(b) Turning function

Figure 4.42 - A sector shaped blob.

(a) Blob
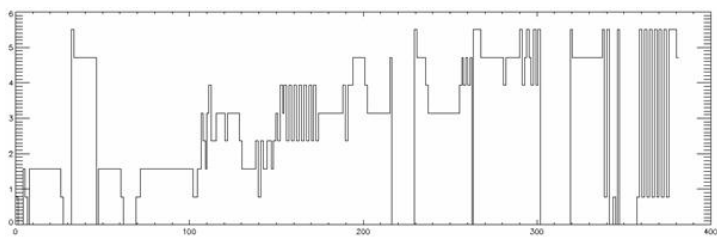
(b) Turning function

Figure 4.43 - A large cross shaped blob.

(a) Blob
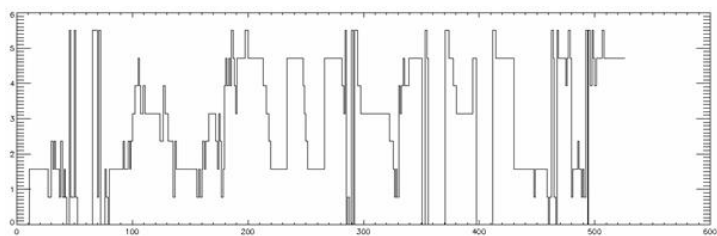
(b) Turning function

Figure 4.44 - A cross shaped blob.

Table 4.2 - Shape indexes 1.

Shape name	$h_1 * per$	$h_2 * per$
circle	50.8	126.9
ellipse	101.3	203.0
square	14.1	35.3
rectangle	39.8	83.3
round square	35.3	80.4
diamond	57.6	142.0
sector	51.9	129.0
large cross	44.6	102.9
cross	54.1	121.2

Table 4.3 - Shape indexes 2.

Shape name	Perimeter	Area
circle	252.5	3543.0
ellipse	384.9	3844.8
square	260.8	3902.0
rectangle	260.2	2907.9
round square	374.0	3804.7
diamond	308.9	3245.4
sector	315.0	3460.9
large cross	381.1	4555.4
cross	525.8	1925.0

Table 4.4 - Shape indexes 3: Complex pattern index (h).

Shape name	h_1	h_2
circle	0.2014	0.5028
ellipse	0.2633	0.5274
square	0.0540	0.1355
rectangle	0.1530	0.3203
round square	0.0943	0.2150
diamond	0.1865	0.4597
sector	0.1649	0.4095
large cross	0.1171	0.2700
cross	0.1029	0.2305

Table 4.5 - Shape indexes 4: Complex pattern index ($L*T/D$).

Shape name	$fracL*T_1D$	h_2
circle	1376.5	1603.4
ellipse	3195.4	5354.4
square	952.4	1017.9
rectangle	2414.8	2714.3
round square	3454.7	4266.0
diamond	2073.6	2758.5
sector	2240.8	2797.0
large cross	3492.9	4494.8
cross	7308.0	10260.2

5 CONCLUSIONS

In this thesis, we have developed a set of new tools and methods for image segmentation and shape analysis. We have proposed modifications on a hue segmentation algorithm to give more options to the user when finding satisfactory methods and parameters that can give output to segments close to the foreground object shapes. We have also proposed ways to reduce the low saturation problem in IHS space for multispectral images and developed and implemented a shape analysis capability inside the segmentation method. We summarize the contents of this thesis in Section 5.1 and make an overview of the contributions, including the papers produced during the thesis in Section 5.2 and provided some new topics derived from this work for future research in Section 5.3.

5.1 Summary

A new approach to make object-based segmentation was presented. It is possible to characterize shapes by the use of turning function. The objective of this method is to show theoretically that it is feasible to work with turning functions and apply them to the classification of shapes and subshapes present in binary images. A complete environment was tailored in order to give access to the shapes present in the images by the use of a hue segmenter.

The shape module needs a segment representing most of the real object, and is able to modify the shape to replace the old one. Considering the turning function space it was shown how to find circles and lines, both represented as a piecewise step function. The mechanism is to make alignments of lines on a turning function transformed space.

Turning functions are sensible to noise effect. This is an important aspect that, in one hand, makes it difficult to promptly find patterns and to manipulate, but in the other, facilitates the identification of any kind of noise, spike and asymmetry.

To make sense the use of turning functions, even being difficult to identify properties in noisy shapes, a set of metric can be used for shape and pixel channel values. It was presented some metrics based on turning functions to enhance the usual metrics. The studies indicate that this feature can help the identification of noises and the classification of the shape by indexes during segmentation for refinement purposes. The adoption of a shape classification allows the creation of new model-based applications for object recognition in images.

117

5.2 Contributions

The contributions of this thesis are:

- to make it possible to include a shape analysis module inside a segmentation process, giving it the tools (i.e., the ability) to manipulate objects during segmentation;

- obtain segments with predefined shapes (e.g., rectangles) as a result of the segmentation process.

- to improve the algorithms presented by Souto (2000) and complement the methodology with several mechanisms;

- to present new ways to make hue segmentation based on multispectral remote sensing imagery;

- to propose a shape analysis approach by the use of turning function.

Some papers were published during the thesis, detailing some important parts. They are listed and briefly commented in the present Section.

"A proposal of object-based image segmentation and turning function to model shapes" (VOLOTAO et al., 2009) introduce the idea of an object-based image segmentation method, presenting some concepts involved and how it is possible to use a shape signature representation to proceed a template-based recognition of the shape.

"Using tangent function to modify image segments" (VOLOTÃO et al., 2010) shows how to compare shapes with turning functions, how to find rectangles when the original shape have noises, and suggests how it is possible to change foreground of segments of rectangular objects in rectangles.

"Using turning functions to refine shapes" (VOLOTAO et al., 2010a) makes an overview of the complete object-based segmentation process, from the image to the new modified image, describing the turning function space and how to make geometric changes in the models. It describes how to make bounding boxes directly in the transformed space.

"Shape characterization with turning functions" (VOLOTAO et al., 2010b) presents definitions and geometric properties related to turning functions in order to proceed

118

the manipulation of the turning function two-dimensional plane space. The focus of this paper is to present a way to identify and modify classes of shapes by operating directly on the turning function space and to identify circles and lines in turning functions to find complex shape classes determined by user defined constraints. It has been shown a number of equations valid for turning functions to make them useful and making possible to find tools to explore the possibilities. Extending the given equations, much more can be developed and used, making the turning function approach helpful for shape analysis.

"Advantages of object-based segmentation to the identification of remote sensing targets" (VOLOTÃO et al., 2010) discusses about some advantages in using an object-based approach in segmentation for the identification of targets.

"Image segmentation refinement by modeling in turning function space" (VOLOTAO et al., 2011) shows the main aspects of the shape analysis module of the object-based segmentation proposal, presenting a brief mathematical background with some geometric operations to replace an interval with a line or a circle. This paper suggest the idea of an active contour by a background field based on the image to drive the proposed modifications. A set of rules drives the results to meet the modeled object as it is supposed (or desired).

"Image segmentation with image and shape phase analysis" (VOLOTÃO et al., 2012) presents the description of the hue segmentation algorithm with hue piecewise-linear remapping, the S-weighted method, some features available in the segmentation algorithm, and a juncture of the hue segmenter with shape analysis on-the-fly, in order to achieve better object shape contour of known object shapes.

5.3 Future Work

We suggest several points to extend the scope of this thesis related to hue segmentation and shape analysis. We briefly mention some possible extensions.

The ability to change the distributions of hue values by piecewise linear transformation was integrated in the algorithm. This tool can be more useful if an automatic way to redistribute the values of hue, considering the angular nature of hue, providing more options to the user to deal with hue statistical distribution. In other words, the above extension can make possible to emphasize selected regions on the feature space according to supervised and non-supervised approaches.

A classification algorithm based on IHS channels (specially the hue and saturation),

with the ability to extract texture mapping, can be useful to make the whole method more complete and flexible. The identification of classes by user intervention or by unsupervised algorithm can make the identification of foreground segments (i.e., objects) easier. For example, if the user could point to a pixel on a roof, the segmentation algorithm could find many other candidate pixels using the same seed and find other segments that behave the same way, making internal analysis on pixel values, texture and shape. The use of indexes, including the suggested indexes based on shape, can make this recognition process feasible.

Our manual turning function modification method was developed for changing manually cuts of the turning function, expecting the user to point any two parts in order to make a "short circuit" between these two ends, replacing any turning function pattern with a new constant function. This kind of operation in turning functions produces a direct link from the two points in the shape, overriding the old noisy pattern to a straight line. The suggestion for future work is the automatic identification of the places that need substitution, e.g. by an entropy analysis of the parts, and the replacement of the noisiest intervals. This approach can be tested with constraints to meet regular shapes (e.g., rectangles, squares, same-length polygons etc).

An upgrade of this last suggestion is our most critical suggestion: the development of a patterns by circles. A circle in turning function is represented by a sloped line, or a regular stair-shaped function, with steps of small size. The turning function length of this pattern is the perimeter of the pattern. If a pattern could be automatically replaced by a well-behaved step-function the achievement of a new level of automatic shape modeling would be effective. A relevant problem is to develop an automatic arc replacement approach, what can be done by starting with the tools proposed in this thesis.

REFERENCES

ARKIN, E. M.; CHEW, L. P.; HUTTENLOCHER, D. P.; KEDEM, K.; MITCHELL, J. S. B. An efficiently computable metric for comparing polygonal shapes. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE Computer Society, Los Alamitos, CA, USA, v. 13, n. 3, p. 209–216, 1991. ISSN 0162-8828. 49

BAATZ, M.; BENZ, U.; DEHGHANI, S.; HEYNEN, M.; HÖLTJE, A.; HOFMANN, P.; LINGENFELDER, I.; MIMLER, M.; SOHLBACH, M.; WEBER, M. et al. **eCognition user guide**. Munich, 2001. 10

BALLARD, D. H.; BROWN, C. M. **Computer vision, 1982**. Upper Saddle River: Prentice Hall, 1982. 548 p. 1, 3, 39

BEUSMANS, J.; HOFFMAN, D.; BENNETT, B. Description of solid shape and its inference from occluding contours. **Journal of the Optical Society of America**, v. 4, n. 7, p. 1155–1167, 1987. 6

BINS, L.; ERTHAL, G.; FONSECA, L. M. G. Um método de classificação não supervisionada por regiões. **Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens**, v. 5, p. 65–68, 1993. 8

CARVALHO, J.; GULIATO, J.; SANTIAGO, S.; RANGAYYAN, R. Polygonal modeling of contours using the turning angle function. In: IEEE, 2007, Vancouver. **Canadian Conf. Elec. Comp. Eng.** Vancouver: CCECE/IEEE, 2007. p. 1090–1093. 9

CAVALLARO, A.; GELASCA, E. D.; EBRAHIMI, T. Objective evaluation of segmentation quality using spatio-temporal context. In: IEEE. **Image Processing. 2002. Proceedings. 2002 International Conference on**. Rochester, New York: IEEE, 2002. v. 3, p. III–301. 103

COELHO, V.; STRAUCH, J.; ESPERANÇA, C. Similarity among multiple geographic representations. In: **Proceedings of 8th GeoWeb**. Vancouver, Canada: GeoWeb, 2009. 56

COOK, M.; PETERSON, B.; DIAL, G.; GIBSON, L.; GERLACH, F.; HUTCHINS, K.; KUDOLA, R.; BOWEN, H. Ikonos technical performance assessment. In: SHEN, S.; DESCOUR, M. (Ed.). **SPIE Proceedings:**

Algorithms for Multispectral, Hyperspectral, and Ultraspectral Imagery VII. Orlando: SPIE, 2001. v. 4381, p. 94–108. 26

DELVES, L.; WILKINSON, R.; OLIVER, C.; WHITE, R. Comparing the performance of sar image segmentation algorithms. **International Journal of Remote Sensing**, Taylor & Francis, v. 13, n. 11, p. 2121–2149, 1992. 76, 102

ESPINDOLA, G.; CAMARA, G.; REIS, I.; BINS, L.; MONTEIRO, A. Parameter selection for region-growing image segmentation algorithms using spatial autocorrelation. **International Journal of Remote Sensing**, Taylor & Francis, v. 27, n. 14, p. 3035–3040, 2006. 1

FENG, W.; PRINET, V.; SONGDE, M. A vector filtering technique for sar interferometric phase images. In: **Proc. IASTED International Symposia, AI**. Innsbruck, Austria: IASTED, 2001. v. 566570. 35

FORSYTH, D.; PONCE, J. **Computer vision: a modern approach**. 2. ed. Upper Saddle River: Prentice Hall, 2011. 792 p. ISBN 978-013608592-8. Available from: <http://books.google.com.br/books?id=gM63QQAACAAJ>. 16

FOSTER, D.; SIMMONS, D.; COOK, M. The cue for contour-curvature discrimination. **Vision research**, Elsevier, v. 33, n. 3, p. 329–341, 1993. 44

FREEMAN, H. Computer processing of line-drawing images. **ACM Computing Surveys (CSUR)**, ACM, v. 6, n. 1, p. 57–97, 1974. 48, 67

FU, K.; MUI, J. A survey on image segmentation. **Pattern recognition**, Elsevier, v. 13, n. 1, p. 3–16, 1981. 13, 78

GONG, Y.; SAKAUCHI, M. Detection of regions matching specified chromatic features. **Computer vision and image understanding**, Elsevier, v. 61, n. 2, p. 263–269, 1995. 7

GONZALEZ, R. C.; WOODS, R. E. **Processamento de imagens digitais**. 3. ed. São Paulo: Pearson Prentice Hall, 2010. 640 p. 15, 78

GULIATO, D.; CARVALHO, J. D. de; RANGAYYAN, R. M.; SANTIAGO, S. A. Feature extraction from a signature based on the turning angle function for the classification of breast tumors. **Journal of Digital Imaging**, Springer, v. 21, n. 2, p. 129–144, 2008. 9

HARALICK, R.; SHAPIRO, L. Image segmentation techniques. **Computer Vision, Graphics, and Image Processing**, Elsevier, v. 29, n. 1, p. 100–132, 1985. 13, 81

IGARASHI, T.; MATSUOKA, S.; KAWACHIYA, S.; TANAKA, H. Interactive beautification: a technique for rapid geometric design. In: ACM (Ed.). **Proceedings of the 10th annual ACM symposium on User interface software and technology**. New York, 1997. p. 105–114. 45

IKONOMATAKIS, N.; PLATANIOTIS, K.; ZERVAKIS, M.; VENETSANOPOULOS, A. Region growing and region merging image segmentation. In: IEEE. **Digital Signal Processing Proceedings, 1997. DSP 97., 1997 13th International Conference on**. Santorini, 1997. v. 1, p. 299–302. ISBN 0-7803-4137-6. 13, 20

JENSEN, J. **Introductory Digital Image Processing**. 3. ed. Upper Saddle River: Prentice Hall, 2004. 4, 16, 30

JEON, H.; TIAN, L.; ZHU, H. Robust crop and weed segmentation under uncontrolled outdoor illumination. **Sensors**, Molecular Diversity Preservation International, v. 11, n. 6, p. 6270–6283, 2011. 14

KORTING, T.; DUTRA, L.; FONSECA, L. G. A resegmentation approach for detecting rectangular objects in high-resolution imagery. **Geoscience and Remote Sensing Letters, IEEE**, IEEE, v. 8, n. 4, p. 621–625, 2011. 9, 20

KORTING, T.; FONSECA, L.; DUTRA, L.; SILVA, F. A new graph-based approach for urban image segmentation. Plzen, Czech Republic, 2008. WSCG 2008. 9

_____. Image re-segmentation applied to urban imagery. **Int Arch Photogram Rem Sens and Spatial Info Sc**, Citeseer, v. 37, p. B3b, 2010. 9, 20

KORTING, T. S. **Um paradigma para re-segmentação de imagens de alta resolução**. 67 p. PhD Thesis (Dissertation: master degree in remote sensing) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2007-12-20 2008. 8, 20

KUMAR, P.; SENGUPTA, K.; LEE, A. A comparative study of different color spaces for foreground and shadow detection for traffic monitoring system. In: IEEE. **Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on**. Singapore, 2002. p. 100–105. 16

LANDSMEER, S. **A study of data-driven and model-driven image segmentation techniques**. Delft, 2007. 1

LES, Z. The processing method as a set of the image transformations in shape understanding. **Computers & Graphics**, Elsevier, v. 25, n. 2, p. 223–233, 2001. 21

LES, Z.; LES, M. **Shape understanding system: the first steps toward the visual thinking machines**. Warsaw: Springer Verlag, 2008. ISBN 978-3540-75768-9. 48

LEW, M. **Principles of visual information retrieval**. London: Springer, 2001. 356 p. (Advances in Pattern Recognition Series). ISBN 9781852333812. 39

LI, Y.; GONG, P. A new generation image interpretation technology based on object-oriented segmentation and classification. In: ASPRS, 2006, San Antonio. **Proceedings...** San Antonio: asprs, 2006. p. 464–468. 10

LIU, W. On-line graphics recognition: state-of-the-art: Graphics recognition. recent advances and perspectives. In: SPRINGER, 2003. **Proceedings...** Heidelberg: Springer, 2004. LNCS, n. 3088, p. 291–304. 45

LLADÓS, J.; VALVENY, E.; SÁNCHEZ, G.; MARTÍ, E. Symbol recognition: Current advances and perspectives. **Graphics Recognition Algorithms and Applications**, Springer, Ontario, p. 104–128, 2002. 45

LONCARIC, S. A survey of shape analysis techniques. **Pattern recognition**, Elsevier, v. 31, n. 8, p. 983–1001, 1998. 44

LUCCA, E. **Avaliação e comparação de algoritmos de segmentação de imagens de radar de abertura sintética**. 207 p. INPE-7507-TDI/721. PhD Thesis (PhD) — Instituto Nacional de Pesquisas Espaciais – INPE, São José dos Campos, 1998. 76, 77, 81, 103

MAKADIA, A.; PAVLOVIC, V.; KUMAR, S. Baselines for image annotation. **International Journal of Computer Vision**, Springer, v. 90, n. 1, p. 88–105, 2010. 14

MARCHAND-MAILLET, S.; SHARAIHA, Y. **Binary Digital Image Processing: A Discrete Approach**. Academic Press, 2000. (Electronics & Electrical). ISBN 978-0124-70505-0. Available from: <http://books.google.com.br/books?id=4WM-Kt5drY0C>. 67

MARTINKAUPPI, J.; SORIANO, M.; LAAKSONEN, M. Behavior of skin color under varying illumination seen by different cameras at different color spaces. In: HUNT, M. (Ed.). **Proceedings...** San Jose: International Society for Optics and Photonics (SPIE), 2001. v. 4301, p. 102–112. 16

MCGARIGAL, K.; CUSHMAN, S.; NEEL, M.; ENE, E. Fragstats: spatial pattern analysis program for categorical maps. **Computer software program produced by the authors at the University of Massachusetts, Amherst**, 2002. 54

MIYAHARA, M. Quality assessments for visual service. **Communications Magazine, IEEE**, IEEE, v. 26, n. 10, p. 51–60, 1988. 7

MOIK, J. G. **Digital Processing of Remotely Sensed Images**. Washington, D.C.: NASA, 1980. 330 p. 14, 29, 37, 61, 95

O'GORMAN, L. Primitives chain code. In: ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, 1988. ICASSP-88., 1988 INTERNATIONAL CONFERENCE ON. **Proceedings...** New York: IEEE, 1988. v. 2, p. 792 –795. ISSN 1520-6149. 67

OTTERLOO, P. V. **A contour-oriented approach to shape analysis**. Herts: Prentice Hall International (UK) Ltd., 1991. 44

PAL, N.; PAL, S. A review on image segmentation techniques. **Pattern recognition**, Elsevier, v. 26, n. 9, p. 1277–1294, 1993. 78

PARK, J. B. Efficient color representation for image segmentation under non-white illumination. In: INTELLIGENT ROBOTS AND COMPUTER VISION XXI: ALGORITHMS, TECHNIQUES, AND ACTIVE VISION. **Proceedings...** Providence: SPIE, 2003. v. 5267, p. 163–174. ISBN 978-0819-45155-2. 1

PAVLIDIS, T. A review of algorithms for shape analysis. **Computer Graphics and Image Processing**, Elsevier, v. 7, n. 2, p. 243–258, 1978. 44

PINHEIRO, A.; GHANBARI, M. Scale space contour approximation for shape similarity evaluation. WSEAS, v. 3, n. 3, p. 818–824, july 2004. ISSN 1109-2750. 9

PRATT, W. **Digital Image Processing: PIKS Scientific Inside**. 4. ed. John Wiley & Sons, 2007. (Wiley-Interscience publication). ISBN 9780471767770. Available from: <http://books.google.com.br/books?id=tpYHZO8111UC>. 67

RANGAYYAN, R.; GULIATO, D.; CARVALHO, J.; SANTIAGO, S. Polygonal approximation of contours based on the turning angle function. **Journal of Electronic Imaging**, v. 17, n. 2, p. 023016, 2008. 9, 39

RUIZ-RUIZ, G.; GÓMEZ-GIL, J.; NAVAS-GRACIA, L. Testing different color spaces based on hue for the environmentally adaptive segmentation algorithm (easa). **Computers and Electronics in Agriculture**, Elsevier, v. 68, n. 1, p. 88–96, 2009. 14

SATELLITE IMAGING CORPORATION. **Pleiades-1B satellite sensor characteristics**. 2013. Available from: <http://www.satimagingcorp.com/satellite-sensors/pleiades-1.html>. 26

SEBARI, I.; HE, D. New object-oriented approach for urban objects extraction from vhsr images. IEEE, p. 4814–4817, 2007. 4

SHIH, F. Y.; CHENG, S. Automatic seeded region growing for color image segmentation. **Image and Vision Computing**, v. 23, n. 10, p. 877 – 886, 2005. ISSN 0262-8856. Available from: <http://www.sciencedirect.com/science/article/pii/S0262885605000673>. 13

SHIH, P.; LIU, C. Comparative assessment of content-based face image retrieval in different color spaces. **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 19, n. 07, p. 873–893, 2005. 16

SOUTO, R. P. **Segmentação de imagem multiespectral utilizando-se o atributo matiz**. 171 p. (INPE-10104-TDI/895). PhD Thesis (Doctoral Thesis in Remote Sensing) — Instituto Nacional de Pesquisas Espaciais–INPE, São José dos Campos, 2000. Available from: <http://mtc-m12.sid.inpe.br/col/sid.inpe.br/jeferson/2003/11.21.10.17/doc>. Access in: 2012. 7, 8, 10, 26, 28, 30, 37, 57, 118

TEIXIDÓ, M.; FONT, D.; PALLEJÀ, T.; TRESANCHEZ, M.; NOGUÉS, M.; PALACÍN, J. Definition of linear color models in the rgb vector color space to detect red peaches in orchard images taken under natural illumination. **Sensors**, Molecular Diversity Preservation International, v. 12, n. 6, p. 7701–7718, 2012. 14

THOMPSON, W.; RIESENFELD, R.; OWEN, J. Determining the similarity of geometric models. In: WORKSHOP ON IMAGE UNDERSTANDING, 1996, Palm Springs. **Proceedings...** Palm Springs: Citeseer, 1996. p. 1157–1160. Access in: 12-feb-2013. 5

UBA, D. M. **Estimação de objetos retangulares em imagens de alta resolução utilizando casamento de modelos**. 78 p. PhD Thesis (PhD) — Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009-05-11 2009. Access in: 28 jan. 2013. 9, 19, 23

UNNIKRISHNAN, R.; PANTOFARU, C.; HEBERT, M. Toward objective evaluation of image segmentation algorithms. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, IEEE, v. 29, n. 6, p. 929–944, 2007. 82

VANTARAM, S.; SABER, E. Survey of contemporary trends in color image segmentation. **Journal of Electronic Imaging**, International Society for Optics and Photonics, v. 21, n. 4, p. 040901–1, 2012. 13

VELTKAMP, R.; HAGEDOORN, M. State of the art in shape matching. In: LEW, M. (Ed.). **Principles of Visual Information Retrieval**. Springer London, 2001, (Advances in Pattern Recognition). p. 87–119. ISBN 978-1-84996-868-3. Available from: <http://dx.doi.org/10.1007/978-1-4471-3702-3_4>. 22

VOLOTÃO, C.; DUTRA, L. V.; SANTOS, R. D. C. d. Image segmentation with image and shape phase analysis. In: FEITOSA, R. Q.; COSTA, G. A. O. P. d.; ALMEIDA, C. M. d.; FONSECA, L. M. G.; KUX, H. J. H. (Ed.). **Proceedings...** São José dos Campos: Instituto Nacional de Pesquisas Espaciais (INPE), 2012. p. 598–601. ISBN 978-85-17-00059-1. Available from: <http://urlib.net/sid.inpe.br/mtc-m18/2012/05.18.14.45>. Access in: 22 abr. 2013. 119

VOLOTÃO, C. F. d. S.; DUTRA, L. V.; SANTOS, R. D. C. d. Uso de função tangente para modificar segmentos de imagens. In: RODRIGUES, R. d. C. M.; ALMEIDA, W. G. d.; ASSIS, T. O.; CHALHOUB, E. S.; CORTIVO, F. D.; FRANÇA, L. F. A.; MACAU, E. E. N.; OLIVEIRA, R. M. d.; PILLAT, V. G.; SANTOS, L. d.; SERPA, D. R.; SILVA, J. D. S. d.; SILVA, M. d. (Ed.). **Anais...** São José dos Campos: Instituto Nacional de Pesquisas Espaciais (INPE), 2010. Available from: <http://urlib.net/sid.inpe.br/mtc-m18/2010/09.30.01.11>. Access in: 22 abr. 2013. 118

VOLOTAO, C. F. d. S.; ERTHAL, G. J.; SANTOS, R. D. C. d.; DUTRA, L. V. Image segmentation refinement by modeling in turning function space. In: ASTOLA, J. T.; KAREN, O. (Ed.). **Image Processing: Algorithms and Systems IX**. Washington: SPIE, 2011. SPIE Proceedings Vol. 7870, p.

7870101–7870111. ISBN 9780819484079. Available from: `<http://spiedigitallibrary.org/proceedings/resource/2/psisdg/7870/1/78701B_1?isAuthorized=no>`. Access in: 22 abr. 2013. 7, 56, 119

VOLOTAO, C. F. d. S.; SANTOS, R. D. C. d.; DUTRA, L. V. Proposta de segmentação de imagens baseada em objetos e uso de função de desvios para modelar formas. In: XII ENCONTRO DE MODELAGEM COMPUTACIONAL, 2009, Rio de Janeiro. **Anais...** 2009. ISBN 9788561433017. Available from: `<http://www.emc2009.iprj.uerj.br/down.php?fid=333>`. Access in: 22 abr. 2013. 118

VOLOTAO, C. F. d. S.; SANTOS, R. D. C. d.; DUTRA, L. V.; ERTHAL, G. J. Using turning functions to refine shapes. In: BARNEVA, R. P.; BRIMKOV, V. E.; M., R.; JORGE, N.; TAVARES, J. M. R. S. (Ed.). **Object Modeling, Algorithms and Applications**. Singapore: Research Publishing Services, 2010. p. 31–44. ISBN 978-981-08-5465-2. Access in: 28 fev. 2013. 44, 109, 118

VOLOTAO, C. F. d. S.; SANTOS, R. D. C. d.; ERTHAL, G. J.; DUTRA, L. V. Shape characterization with turning functions. In: 17TH INTERNATIONAL CONFERENCE ON SYSTEMS, SIGNALS AND IMAGE PROCESSING, 2010, Rio de Janeiro Niterói. **Proceedings...** Editora da Universidade Federal Fluminense, 2010. v. 1, p. 554–557. ISBN 9788522805655. Available from: `<http://www.creacteve.com.br/iwssip/nav/papers/paper_188.pdf>`. Access in: 28 fev. 2013. 39, 118

VOLOTÃO, C. F. S.; DUTRA, L. V.; ERTHAL, G. J.; SANTOS, R. D. C. Vantagens da segmentação baseada em objetos para a identificação de alvos em imagens de sensores remotos. In: 4º ENCONTRO DE USUÁRIOS DE SENSORIAMENTO REMOTO DAS FORÇAS ARMADAS, 2010, São José dos Campos. **Proceedings...** São José dos Campos: IEAv, 2010. 119

WOLFSON, H. On curve matching. **Pattern Analysis and Machine Intelligence, IEEE Transactions on**, IEEE, v. 12, n. 5, p. 483–489, 1990. 49

ZHANG, D.; LU, G. Review of shape representation and description techniques. **Pattern recognition**, Elsevier, v. 37, n. 1, p. 1–19, 2004. 44

ZHANG, Y. **Image Engineering: Processing, Analysis, and Understanding**. Singapore: Cengage Learning, 2009. 714 p. ISBN 9789814239639. 1, 21, 22, 39

ZHANG, Y. J. A survey on evaluation methods for image segmentation. **Pattern recognition**, Elsevier, v. 29, n. 8, p. 1335–1346, 1996. 77