UNIVERSIDAD DE GUANAJUATO

CAMPUS IRAPUATO-SALAMANCA

DIVISIÓN DE INGENIERÍAS

Inteligencia Computacional para la Segmentación de Imágenes Naturales

TESIS

Que para obtener el grado de:

Doctor en Ingeniería Eléctrica

PRESENTA:

M.I. Rocío Alfonsina Lizárraga Morales

Director:

Dr. Raúl E. Sánchez Yáñez

co-Director:

Dr. Víctor Ayala Ramírez

SALAMANCA, GTO. Febrero, 2014

Acknowledgements

I would like to thank everyone who helped and supported me throughout my doctoral studies.

Foremost, I want to express my greatest gratitude to my advisor Dr. Raul Sanchez-Yanez, for his continuous support and motivation during the years I have been under his guidance. I would like to thank him for encouraging my research and for allowing me to grow as a research scientist and as a person. I would also like to thank Dr. Victor Ayala-Ramirez, for all his insightful comments and for all his scientific and personal advices.

My sincere thanks are also to the review committee members, Dr. J. Gabriel Aviña Cervantes, Dr. Mario Ibarra Manzano, Dr. Jean-Bernard Hayet and Dr. Michel Devy. Thank you for your time, suggestions and feedback in the improvement of this manuscript.

I would like to thank Prof. Matti Pietikäinen, for making possible my research visit to the Center for Machine Vision Research and for all his support, guidance and warm welcome to Finland. Also, I thank Dr. Guoying Zhao and MEng. Yimo Guo, for all their support, their valuable comments to my research and for being a great influence as research scientists.

On a personal note, I want to thank my parents Norma Morales and Arturo Lizárraga and my brother Omar Lizárraga, for all their support, love and guidance throughout

my life. This thesis is dedicated to them.

Additionally, I would like to thank all my lab mates in the LaViRIA, for all the good moments, feedback, talks and coffee breaks. Especially, I would like to thank Uriel, Jonathan, Fernando and Patlán.

My heartfelt gratitude also goes to Luis, for his patience and love during the final stage of this research. Finally, I would like to gratefully thank my dear friends: Ivon, Elena, Irving, Miyuki, Yesi, Pita, Pepe, Ana, Hugo and Elisa. Thank you for your continuous encouragement and friendship.

Institutional Acknowledgements

I would like to acknowledge all those institutions that made possible the accomplishment of the research reported in this manuscript.

- To the Universidad de Guanajuato for all the academic and personal support. Also, I would like to thank the Dirección de Apoyo a la Investigación y al Posgrado (DAIP) for the financial support through the grant *Formación de Jóvenes Investigadores* (NUA 143319).

UNIVERSIDAD
DE GUANAJUATO

- To the National Council of Science and Technology (CONACyT) for the grant provided with number **206622**.

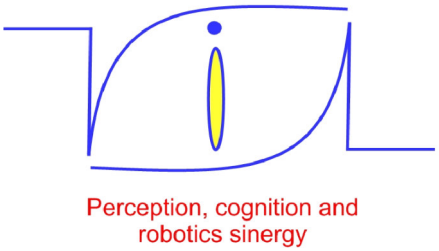
- To the Finnish Center for International Mobility (CIMO), for the grant provided during the research visit to the University of Oulu.

- To the Center for Machine Vision Research and the University of Oulu, for all the support and guidance during the research visit.

UNIVERSITY of OULU
OULUN YLIOPISTO

- To the Consejo de Ciencia y Tecnologa del Estado de Guanajuato (CONCyTEG) for the financial support at the beginning of my doctoral program.

- To the Laboratorio de Visión, Robótica e Inteligencia Artificial (LaViRIA), for all the technical and personal support.



Contents

List of Figures

List of Tables

Preface

Image segmentation is a central topic in computer vision research with a long tradition as one of the fundamental problems. Its importance is based on its use as a pre-analysis of images in the development of high-level tasks, such as object recognition, tracking, scene understanding, image retrieval, just to mention a few. The performance of these aforementioned high-level tasks largely depends on the accuracy and robustness of the image segmentation method.

Segmentation refers to the process of partitioning a digital image into multiple regions. The partition consists in assigning a label to every pixel within an image, in such a way that pixels with the same label are homogeneous in a set of particular features, and in addition, are spatially connected. The main motivation for image segmentation is to provide a compact representation of data, wherein all subsequent processing can be done at a region level, instead of at a pixel level.

A considerable number of features can be taken into account during the partition process, e.g. gray-level, color, texture, shape, depth. Simple algorithms might be able to separate regions with features without variations. However, they might have difficulties when dealing with regions with uneven features. This is the case of natural images, where conditions of non-uniform illumination, noise, feature inhomogeneities and general uncertainties are always present. Although image segmentation is one of the fundamental problems within computer vision research, it still represents a challenging

task due to the difficulty in the management of such conditions.

Computational intelligence is a set of nature-inspired methodologies to address the design of intelligent systems, which are tolerant to imprecision, uncertainty, partial truth and approximation. Such methodologies have been developed in order to achieve tractability, robustness and low-cost solutions. Computational intelligence primarily includes methodologies like artificial neural networks, evolutionary computation, fuzzy logic and rough sets. All these techniques have been successfully employed for various image processing tasks, including image segmentation, enhancement and classification, both individually or in combination with other computational intelligence techniques. The use of these methodologies for image segmentation tasks has a growing interest, since they are able to address conditions that classical hard computing techniques cannot.

In this study, we propose two image segmentation approaches based on computational intelligence elements, specifically on rough set's elements. In our methods, we aim to represent the pixels that are similar to their neighbors, resulting in a description tolerant of feature variations and noise. The first approximation presented in this study is an improvement to the existing rough set-based methods using only color cues. In the second approach, the integration of texture cues is proposed. These improvements allow our method to overcome some performance issues shown by prior rough set-based approaches. The method is evaluated through a set of qualitative and quantitative tests over a comprehensive database, showing that the proposed approach produces high-quality segmentation outcomes, better than those obtained using previous rough set-based and state-of-the-art segmentation approaches.

Structure of this dissertation

This dissertation is organized in two parts, which comprises the two image segmentation approaches proposed in this study. The first part includes **Chapters 1** to **3**. In **Chapter 1**, an introduction to the thesis subject is provided. Additionally, the image segmentation problem is defined in detail and the most common segmentation evaluation frameworks are described. Moreover, an overview of the use of computational intelligence approaches to image segmentation tasks is presented. In **Chapter 2**, the background of the rough set theory and its use in image segmentation, is reviewed. In **Chapter 3**, an improvement to a previously proposed rough-set-based segmentation approach using an adaptable threshold selection and perceptual color spaces, is introduced. Here, a number of issues detected in the previously proposed approach are addressed.

The second part of this dissertation comprises the addition of texture features to the rough-set-based segmentation approach. Hence, a comprehensive review of the texture analysis is first presented in **Chapter 4**. In this chapter, an overview of the methods for texture extraction and texture representation is presented. Additionally, the fundamental problems of texture analysis, like classification, synthesis and segmentation of texture, are reviewed. In **Chapter 5**, the integration of color and texture cues as an improvement of the segmentation approach, is presented. Finally, in the **General Conclusions**, the main results and contributions obtained in the thesis are summarized, and besides themes for further investigation are proposed.

Publications resulting from this research

During the research process of this study, different computational intelligence techniques and image features were studied. The publications resulting from this research are enumerated below:

1. R. A. Lizarraga-Morales, R.E. Sanchez-Yanez, V. Ayala-Ramirez. Periodicity and texel size estimation of visual texture using entropy cues, Computación y Sistemas. 14(3): 309-319 (2011).

2. R. A. Lizarraga-Morales, R.E. Sanchez-Yanez, V. Ayala-Ramirez. Homogeneity cues for texel size estimation of visual texture. J. F. Martinez-Trinidad et al. (Eds.): MCPR 2011, LNCS 6718, pp. 220-229, 2011.

3. R. A. Lizarraga-Morales, R.E. Sanchez-Yanez, V. Ayala-Ramirez, Visual texture classification using fuzzy inference, Proc. Tenth Mexican International Conference on Artificial Intelligence (MICAI 2011), pp. 150-154, Puebla, Mexico, Nov 26-Dec 04, 2011.

4. R. A. Lizarraga-Morales, Y. Guo, G. Zhao and M. Pietikäinen. Dynamic Texture Synthesis in Space with a Spatio-Temporal Descriptor. J.-I. Park and J. Kim (Eds.): ACCV 2012 Workshops, Part I, LNCS 7728, pp. 38-49, 2013.

5. R. A. Lizarraga-Morales, R. E. Sanchez-Yanez and V. Ayala-Ramirez. Fast texel size estimation in visual texture using homogeneity cues. Pattern Recognition Letters. 34(1):414-422. (2013).

6. R. A. Lizarraga-Morales, R. E. Sanchez-Yanez, V. Ayala-Ramirez and A. J. Patlan-Rosales. Improving a rough set theory-based segmentation approach using adaptable threshold selection and perceptual color spaces. Accepted to be published in Journal of Electronic Imaging (SPIE).

7. R. A. Lizarraga-Morales, R. E. Sanchez-Yanez, V. Ayala-Ramirez and F. E. Correa-Tome. Integration of Color and Texture Cues in a Rough-Set-based Segmentation Method. Accepted to be published in Journal of Electronic Imaging (SPIE).

It deserves particular attention the work developed in the texture analysis field, which include the articles published during the first two years of this doctoral program. Such publications were developed while exploring the fundamental problems of texture analysis, the use of different texture features for image segmentation and the computational intelligence approaches. The conducted research on this field, certainly helped in the further development of this study. Furthermore, we would like to highlight the dynamic texture synthesis approach studied during a research stay at the Center for Machine Vision Research at the University of Oulu, Finland, under the supervision of the renowned Prof. Matti Pietikäinen.

Part I

Image Segmentation

CHAPTER 1

Introduction

Image segmentation has been a central problem in computer vision and pattern recognition for many years. The goal of segmentation is to simplify the representation of an image into something that is more meaningful and easier to analyze. Segmentation refers to the process of partitioning a digital image into homogeneous regions [29], that are assumed to correspond to significant objects in the scene.

The partition consists in separating regions using a feature homogeneity criterion and a spatial connection norm. This means that if two regions share the same features, but they are disjoint; they are considered as two different segments. This is, in fact, the difference between segmentation and clustering. According to Haralick and Shapiro [48], the difference is that in clustering, the grouping is done in the feature space, while in segmentation, the grouping is done in the spatial domain of the image. The segmentation task can be equivalently achieved by finding the boundaries between the regions. These two strategies, region-based and edge-based, have been proven to be equivalent just in the case that the boundaries are closed [78]. In order to concentrate all these concepts, we can say that the main motivation for image segmentation is to

provide a compact representation of data, wherein all subsequent processing can be done at a segment level. This preprocessing results in significant computational gains and an improvement in the understanding of the image content. Segmentation is mainly employed as a preprocessing to enhance, analyze, categorize and concentrate information from images.

1.1 Image Segmentation: Problem definition

The segmentation problem has been formally defined by Pal and Pal [98] as: if F is the set of all pixels and $P()$ is a uniformity (homogeneity) predicate defined on groups of connected pixels, the segmentation is a partitioning of the set F into a set of connected subsets of regions $(S_1, S_2, \ldots, S_n)$ such that

$$\bigcup_{i=1}^{n} S_i = F \text{ with } S_i \bigcap S_j = \emptyset, i \neq j. \tag{1.1}$$

The uniformity predicate $P(S_i) = true$ for all regions (S_i) and $P(S_i \cup S_j) = false$, when S_i is adjacent to S_j.

According to Haralick [45], a good segmentation must have certain characteristics:

- Regions of a segmented image should be uniform and homogeneous with respect to some characteristic such as intensity, gray tone, color or texture.
- The interior of each region should be simple and without small holes.
- Adjacent regions of segmentation should have significantly different values with respect to the characteristic on which they are uniform.
- Boundaries of each segment should be smooth, not ragged, and must be spatially accurate.

The fulfillment of all these characteristics in a simultaneous manner represents a major challenge and due to their complexity, a number of methods has been developed in

addressing the problem of image segmentation. The main approximations are reviewed in the next section.

1.2 Image segmentation: Literature review

In this section, the state-of-the-art of this research field is reviewed and discussed. The objective is to develop a useful separation of segmentation algorithms and to delineate the scope of the present study.

A considerable number of approaches has been proposed in order to address the problem of image segmentation. Due to the large number of methods and their technical differences, some authors have focused their efforts to categorize them. From a high-level perspective, Vantaram and Saber [136] have categorized segmentation procedures centered in the final application. This categorization may be carried out according to: (1) the human interaction, (2) the nature of the media and (3) the number and type of attributes used. The first criterion discriminates the approaches that require human intervention from those that operate fully unsupervised. The second criterion separates approaches depending on whether these methods are intended to be used on video sequences or still images. Finally, the last criterion categorizes methods according to the features used for the association of pixels, e.g. color, texture or a suitable combination. The scope of the literature review presented here is limited to the features used, yet it is focused on methods that are fully unsupervised and that are applied for static images.

The early surveys by Fu and Mui [35], Haralick and Shapiro [48] and Pal and Pal [98] were mainly focused on the definition of the segmentation problem and the first approximations developed for the treatment of monochromatic images. Other specialized surveys for a given use of features have been also proposed. Surveys for color-based image segmentation have been presented by Cheng et al. [14], Lucchese and Mitra [78], and the more recent work by Vantaram and Saber [136]. A survey for the use of texture features is the presented by Reed and Dubuf [107]. In the same way, the integration of

color-texture descriptors is reviewed by Ilea and Whelan [53].

Depending on their technical grounding, the segmentation methods may be also separated into two groups: spatially-guided and spatially-blind methods. The main idea of the spatially-guided approaches is that pixels that are neighbors, may belong to the same segment or group. Hence, the goal is to agglomerate adjacent pixels. Their main drawback is that, even when the resulting segmented regions are spatially well-connected and compact, there is no guarantee that all pixels in a segment are homogeneous in a specific feature space. Moreover, sequential design (pixel-by-pixel agglomeration) of these procedures, often results in intensive computational schemes with significant memory requirements. Among these approaches, the quality of the segmentation is dependent on the initial seeds' selection, and on the homogeneity criteria used. The spatially-blind algorithms assume that the features on the surface of an object are unvarying and therefore, the object will be represented as a cluster of points in the given feature space. Because of their simplicity and low computational cost, this kind of methods have been widely adopted in the development of segmentation algorithms. Different methods in both spatially-guided and spatially-blind approximations have been proposed based on the use of a variety of features or combination of them. The most-used features are discussed below.

1.2.1 Color and intensity-based approaches

Color information is the most common feature used to determine similarity between pixels in an image. This is because color data is directly available from the pixels. As it was mentioned before, following their technical basis, these methods can be separated in spatially-guided and spatially-blind approaches. On one hand, among the spatially-guided image segmentation methods, it is possible to find two main tendencies: region-based and energy-based techniques. The region-based methods, typically employ techniques involving region growing [32, 24, 137, 33] and split-and-merging. The region growing approach starts with the definition of a pixel, also called seed. Then,

pixels around the seed are accumulated, based on a color homogeneity criterion. The growth stops when no pair of pixels satisfy the color similarity. The split-and-merge techniques consist in repetitively splitting the image until all the segments satisfy the particular color uniformity criterion. After the splitting, an additional region merging is carried out in order to fuse neighboring subregions that may belong to the same object in the scene. The energy-based techniques aim to minimize energy cost functions. Among this kind of methods, we can find the active contours approximations [58, 105] and the graph-based segmentation techniques [118, 82]. The active contours attempt to minimize an energy associated to a given contour as a sum of an internal and external energy. Among the graph-based methods, an image is considered as a graph, where the pixels are nodes. The segmentation is performed by a max-flow/min-cut optimization.

On the other hand, the spatially-blind methods perform segmentation in a given color representation ignoring the spatial relationship between pixels. Examples of these approaches include clustering [56, 13] and histogram-based approximations [64, 15]. Clustering basically views each pixel intensity as a point within a cloud in an n-dimensional feature space. The main idea is to analyze the cloud of points to identify meaningful pixel groups or clusters by separating the cloud using predefined metrics or objective functions. One of the most popular approaches is the use of the k-means algorithm [75, 87], which is an iterative technique that is used to partition an image into K clusters. The k-means algorithm is guaranteed to converge, but it may not return the optimal solution. Moreover, the quality of the solution depends on the initial set of clusters and the value of K. Another popular method in clustering is the Mean Shift approach, which is a non-parametric feature-space analysis technique, also called as mode seeking algorithm. The Mean Shift procedure was originally presented by Fukunaga and Hostetler [36] for pattern recognition applications. However, its usage for image segmentation was introduced by Comaniciu and Meer [20]. In this method, the clusters are places in the feature space where data points tend to be closer. Assuming that the data are samples from a probability distribution, the centers of the clusters are located

on the local maxima.

The histogram-based methods are established on the principle that each object in the image may be identified as a mode in the histogram. These methods offer some advantages over the clustering approximations. This is because in the histogram-based methods, there is no need for a-priori information about the image, such as the number of classes or clusters, or the palette of colors to be used. In this technique, the representative objects within the scene are identified as significant peaks in the intensity histogram. Depending on the number of peaks, a set of thresholds is established and a multithreshold segmentation is carried out. Disadvantages of this approximation include the sensitivity to noise and intensity variations, the difficulty to identify significant peaks in the histogram, and the fact that the regular histogram ignores the spatial relation between neighboring pixels. In order to address the problems of the histogram-based methods, Mohabey and Ray [89] introduced the concept of histon, based on the rough set theory [101]. More details about this methodology and further improvements are detailed in Chapters 2 and 3.

Concerning the color-features representation, it is known that the performance of a color-based segmentation method highly depends on the choice of the color space [6]. The RGB color space is the most used in the literature for image segmentation tasks. A particular color in the RGB space is specified in terms of the intensities of three additive colors: red, green and blue [31]. Although the RGB space is the most used, this representation does not permit the emulation of the higher level processes which allow the perception of color in the human visual system [78].

Different studies have been oriented to the determination of the best suited color representation for a given segmentation approach [43, 7, 6, 21, 117]. Some of them have found that the so-called perceptual color spaces, e.g. $CIELab$ and $CIELuv$, are the most appropriate when the resemblance to the human visual system is desirable. The main advantage of the perceptual color representations is that the Euclidean distance between two points is proportional to the difference perceived by a human between the

two colors represented by such points. This ability to express color difference of human perception by Euclidean distance is very important [14] because any direct comparison between colors can be performed based on their geometric separation.

1.2.2 Texture-based methods

Texture segmentation basically involves the identification of regions with the same texture features, so that further analysis can be performed on the respective regions alone. An effective texture segmentation algorithm is very useful in many application areas, such as analysis of remote sensing images, industrial monitoring of product quality, medical image analysis and image retrieval.

Different approaches to image segmentation using textural features have been proposed. Early methods proposed for unsupervised region-based texture segmentation include approaches based on split-and-merge methods [12], pyramid node linking [104], selective feature smoothing with clustering [23], and a quad-tree method combining statistical and spatial information [122]. Examples of more recent approaches are methods based on local binary patterns for both static [95, 106] and dynamic textures [10], Markov random field models [114], structure-based segmentation [127, 84] and active contours [112, 113].

1.2.3 Color-texture-based approximations

As it was mentioned before, there is a significant number of features that may be considered during the segmentation process, such as gray-level, color, motion, texture, etc. However, the task of finding a single feature to describe an image content may be difficult. A significant amount of proposals have been dedicated to the development of segmentation algorithms where color or texture are analyzed alone. However, it has been found that humans often combine multiple sensory cues to improve their perceptual performance [110]. This fact has motivated recent research to focus on the integration of more than one feature. In particular, the integration of color and texture cues has

shown to be strongly linked to the human perception [53].

Recently, Ilea and Whelan [53] have categorized such segmentation methods according to the approach used for the extraction and integration of color and texture features. Three major trends have been identified: (1) Implicit color-texture integration, where the texture is extracted from one or multiple color channels. (2) Extraction of features in sequence and (3) extraction of color and texture features on separate channels and their combination in the segmentation process. According to the authors, the approaches that extract cues in separate channels have the advantage of optimizing the contribution of each feature in the segmentation algorithm.

Depending on their technical basis, the segmentation methods that combine color and texture may be also separated into the two groups previously mentioned: spatially-guided and spatially-blind methods. Examples of the spatially-guided approaches include the split-and-merge [93, 11], region growing [37, 33], watershed [3, 86] and energy minimization [44, 60]. For these approaches, the quality of the segmentation depends on the initial seeds selection and on the homogeneity criteria used. On the other hand, examples of spatially-blind approximations include the clustering methods [88, 97, 52]. Because of their simplicity, this kind of methods have been widely adopted in the development of color-texture segmentation algorithms. However, it results difficult to adjust the optimal number of clusters and their initialization for different images. Therefore, the color-texture information may not be optimally evaluated during the space partitioning process.

1.3 Performance evaluation of segmentation algorithms

Although the characteristics for a good segmentation have been established, image segmentation is often viewed as an ill-defined problem with no perfect solution but multiple, generally acceptable solutions [136]. An objective evaluation performance of a given algorithm depends on the final application. Computational efficiency or stability

may be important for some applications, while for other applications the resemblance to the human perceptual system may be desirable. A usual practice for the evaluation is to show a small set of the resulting images and to indicate why the outcomes are good. Due to the subjectivity of this evaluation methodology, different approaches have been proposed.

1.3.1 Segmentation Benchmark

Taking into account the subjectivity of the evaluation frameworks and the lack of a standard database, a segmentation benchmark, named Berkeley Segmentation Data Set and Benchmark (BSD), was developed in the University of California at Berkeley [83]. Such a database consists of 300 images of 481×321 pixels size, and it is an empirical basis for the evaluation of segmentation algorithms. For each image, a set of 5 to 9 reference segmentations are available (a total of 1633 manual segmentations) and can be used to quantify the reliability of a given method. Furthermore, the diversity of content in this data set, that includes landscapes, animals, buildings and portraits, makes it a challenge for any segmentation algorithm. Samples of the original images and of the manually generated regions can be seen in Fig. 1.1.

A thorough analysis of the obtained ground truth from the BSD has yielded to the conclusions [83] that (1) an arbitrary image may have a unique suitable segmentation outcome while others possess multiple acceptable solutions and (2) the variability in adequate solutions is primarily due to differences in the level of attention (or granularity) and the degree of detail from one human observer to another. Therefore, segmentation algorithms must aim to provide a generally acceptable outcome rather than a universal solution.

Once the benchmark segmentation database was established, different quantitative measures were proposed in order to evaluate segmentation algorithms in an objective manner. Quantitative measures are used to evaluate how well a segmentation algorithm imitates human perceptual grouping, thus they are used in conjunction with the BSD.

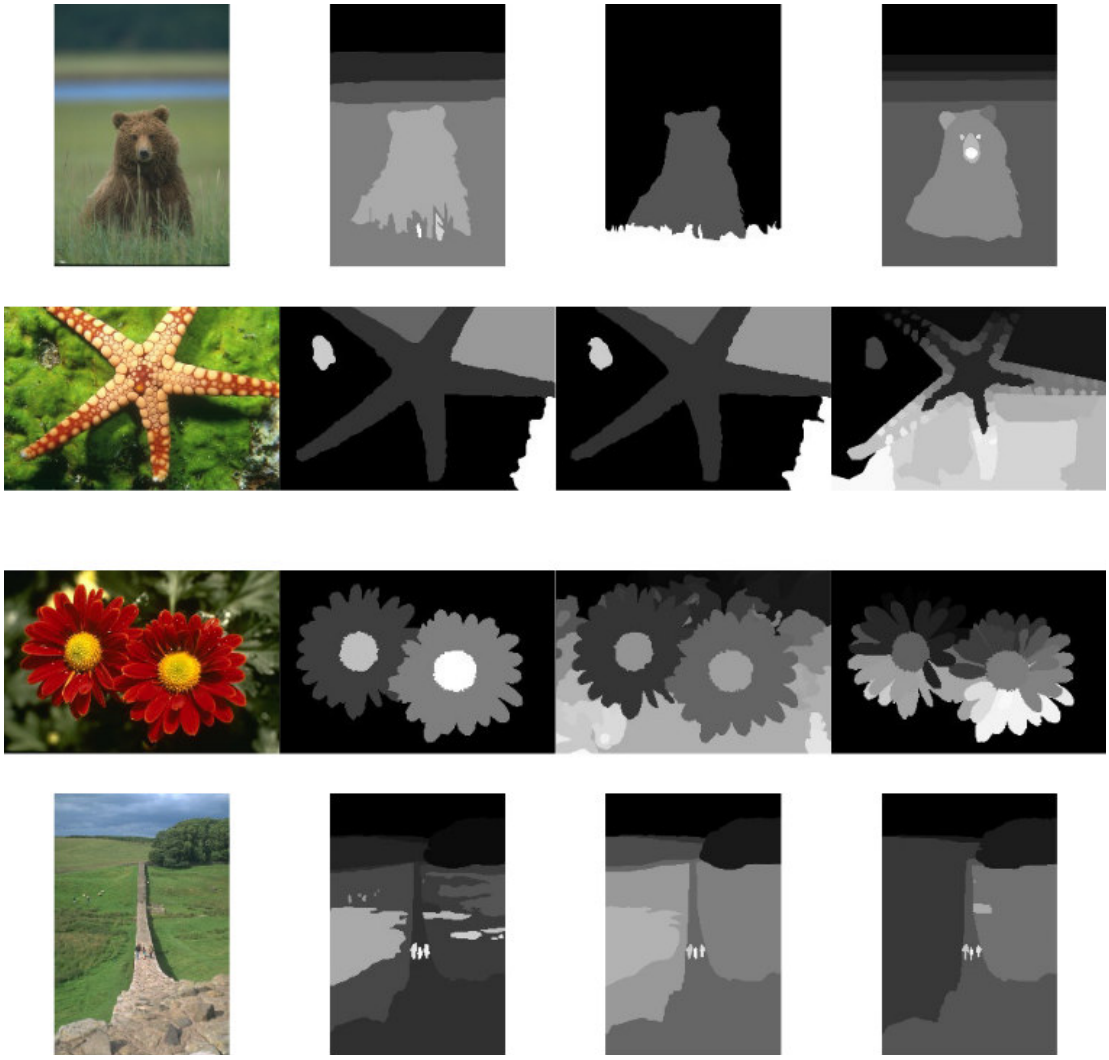


Figure 1.1: Four examples out of the 300 of the Berkeley Segmentation Benchmark. Original images and three examples of the manually generated segmentations.

In the context of this task, Unnikrishnan [133] established a set of requirements for a measure of segmentation correctness:

- Non-degeneracy: It does not have degenerate cases where unrealistic input instances give abnormally high values of similarity.
- No assumptions about data generation: It does not assume equal number of the labels or region sizes in the segmentations.
- Adaptive accommodation of refinement: The term refinement denotes the degree of detail in the segmentation of a given image. A given measure must include the differences in granularity at which the image is perceived by different observers.
- Comparable scores: A given measure provides scores that permit meaningful comparison between segmentations of different images and between different segmentations of the same image.

According to these aforementioned requirements, three metrics were selected in our study for segmentation algorithms assessment: the Boundary Displacement Error (BDE)[34], the Global Consistency Error (GCE) [83] and the Probabilistic Rand Index (PRI) [132].

1.3.2 The Boundary Displacement Error (BDE)

The **BDE** [34] is intended to evaluate the segmentation quality in terms of the precision of the extracted region boundaries. The BDE measures the average displacement error between the boundary pixels of the segmented image and the closest boundary pixels in the set of ground truth segmentations. Let B represent the boundary points set derived from the segmentation and G^B the boundary ground truth. Two distance distribution signatures are used, one from the ground truth to the estimated segmentation, denoted by D_G^B, and the other one is from the estimated segmentation to the ground truth, denoted by D_B^G. A distance error from a set B_1 to a set B_2 of boundary points, denoted by

$D_{B_1}^{B_2}$, is a discrete function whose distribution characterizes the discrepancy, measured in distance, from B_1 to B_2. The distance from an arbitrary point x in set B_1 to B_2 is defined as the minimum distance from x to all the points in B_2, as it is seen in Eq. 1.2,

$$d(x, B_2) = \min d_E(x, y), \forall y \in B_2, \tag{1.2}$$

where d_E denotes the Euclidean distance between points x and y. The discrepancy between two boundary images B_1 and B_2 is described by Eq. 1.3,

$$D_{B_1}^{B_2} = \frac{1}{n} \sum_{x \in B_1} d(x, B_2), \tag{1.3}$$

where n is the number of pixels in the image B_1. The final BDE is computed with the mean between $D_{B_1}^{B_2}$ and $D_{B_2}^{B_1}$. A BDE with a near-zero mean and a small standard deviation indicates a high quality of the segmentation outcome.

$$BDE = (D_{B_1}^{B_2} + D_{B_2}^{B_1})/2. \tag{1.4}$$

1.3.3 Global Consistency Error (GCE)

The **GCE** [83] is designed so that when comparing two different segmentations, if one is a refinement of the other, the error value should be very small, or even zero. By refinement, the authors mean that the segmentations are consistent, but one segmentation has a higher level of detail than the other. The GCE takes two segmentations S_1 and S_2 as inputs, and produces a real-valued output in the range $[0, 1]$, where zero means no error. For a given pixel p_i, consider the regions in S_1 and S_2 that contain such pixel. The regions are sets of pixels. If one segment is a proper subset of the other, a pixel lies in an area of refinement and the local error should be zero. If there is no subset relationship, then, the two regions overlap in an inconsistent manner. Let $\setminus$ denote set difference and $|x|$ be the cardinality of set x. If $R(S, p_i)$ is the set of pixels corresponding

to the region in segmentation S that contains the pixel p_i, the local refinement error is defined in Eq. 1.5.

$$E(S_1, S_2, p_i) = \frac{|R(S_1, p_i) \backslash R(S_2, p_i)|}{|R(S_1, p_i)|}. \tag{1.5}$$

where $R(S_1, p_i) \backslash R(S_2, p_i) = \{p \in R(S_1, p_i) | p \notin R(S_2, p_i)\}$. Note that this local refinement error is not symmetric $E(S_1, S_2, p_i) \neq E(S_2, S_1, p_i)$. It encodes the error in one direction only. In order to force all the local refinements to be in the same direction, the Global Consistency Error es computed with Eq. 1.6,

$$GCE(S_1, S_2) = \frac{1}{n} \min \left[\sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i) \right], \tag{1.6}$$

where n is the number of pixels in the image.

1.3.4 The Probabilistic Rand-Index (PRI)

The **PRI** [132] counts the number of pairs of pixels whose labels are consistent both, for the ground truth and for the segmentation result. Consider a set of manual segmentations (ground-truth) $\{S_1, S_2, \ldots, S_k\}$ of an image $X = \{x_1, \ldots, x_N\}$ consisting of N pixels. Let S_{test} be the segmentation that is to be compared with the ground truth. The label of the point x_i in S_{test} is denoted by $l_i^{S_{test}}$ and in the manually segmented image S_k it is denoted by l_i^k.

In order to model the label relationships for each pixel pair, the underlying distribution is estimated. This may be visualized as the scenario where each human segmenter provides information about the segmentation S_k of the image in the form of binary numbers $I\left(l_i^{S_k} = l_j^{S_k}\right)$ for each pair of pixels (x_i, x_j). The set of all perceptually correct segmentations defines a Bernoulli distribution, giving a random variable with expected value denoted p_{ij}. Hence, the set $\{p_{ij}\}$ for all unordered pairs (i, j) defines a generative model of correct segmentations for a given image. In [132], the authors propose the choice of estimator for p_{ij}, as the mean of the corresponding Bernoulli distribution as

given by:

$$p_{ij} = \frac{1}{K} \sum_{k} I(l_i^{S_k} = l_j^{S_k}). \tag{1.7}$$

The PRI is finally defined as:

$$PR(S_{test}, \{S_k\}) = \frac{1}{\binom{N}{2}} \sum_{i,j;i<j} \left[c_{ij} p_{ij} + (1 - c_{ij})(1 - p_{ij}) \right], \tag{1.8}$$

where c_{ij} denotes the event of a pair of pixels i and j having the same label in the test image S_{test}:

$$c_{ij} = I\left(l_i^{S_{test}} = l_j^{S_{test}}\right) \tag{1.9}$$

This measure takes values in $[0, 1]$, where 0 means that the S_{test} and $\{S_1, S_2, \ldots, S_k\}$ have no similarities, and a PRI value of 1 means that all segmentations are identical. This measure is considered as the most important in our evaluation framework because, as it is pointed out by Yang [138], there is a good correlation between the PRI and the human perception through the hand-labeled segmentations.

1.4 Computational intelligence in image segmentation tasks

Real-world problems are ill-defined, they are difficult to model and usually require large-scale solution spaces. Computational intelligence (CI) is a set of methodologies to address the design of intelligent systems which, in contrast to classical hard computing techniques, are tolerant to imprecision, uncertainty, partial truth, and approximation [80]. Such methodologies were developed in order to achieve tractability, robustness, and low-cost solutions. The guiding principle is to devise methods that lead to an acceptable solution at a low cost, by seeking for an approximate solution to an imprecisely formulated problem.

CI is, in general, the symbiotic use of many emerging computing disciplines. CI primarily includes Artificial Neural Networks, Evolutionary Computation, Fuzzy Logic and Rough sets. All these techniques have been successfully employed for various image processing tasks including image segmentation, enhancement and classification, both individually or in combination with other CI techniques. Given the importance of the rough sets in this study, their background, definition and more details will be presented in the next chapter. The remaining methodologies are introduced in the following subsections.

1.4.1 Fuzzy Logic

Fuzzy logic, firstly introduced by Zadeh [140], provides a language with syntax and local semantics, in which we can perform the qualitative knowledge of a human expert about the problem to be solved [115]. Through fuzzy sets, it is possible to represent and process linguistic information. Fuzzy sets are inherently inclined to deal with linguistic knowledge and produce more interpretable solutions. Fuzzy sets allow the tolerance to ambiguity, uncertainty and imprecision, facilitating the development of complex systems.

Fuzzy logic has been extensively used in many application fields due to its characteristic advantages. Specifically in image segmentation tasks, fuzzy logic gives a mechanism to represent the ambiguity within an image [115]. Fuzzy segmentation approaches include the use of fuzzy clustering with the fuzzy c-means algorithm [1, 68, 17, 124], fuzzy rule-based approaches [57], measures of fuzziness [79], detection of moving objects [16], among others.

1.4.2 Artificial Neural Networks

The Artificial Neural Networks (ANNs), first explored by Rosenbaltt [108], are computational structures to learn patterns from examples. They have been useful in the representation of non-linear phenomena between multiple variables. Using a training

set that samples the relation between inputs and outputs, and a back-propagation algorithm, ANNs give a supervised learning algorithm that performs local optimization.

The main contributions of ANNs in the field of image analysis include classification and clustering. Examples of their application are, for example, the one proposed by Ji and Park [55] that uses a Self Organization Map (SOM) in a watershed algorithm. Other example is the one introduced by Ilea and Whelan [52], which uses a SOM for color quantization. More examples may be consulted in [115].

1.4.3 Evolutionary Computation

The evolutionary computation comprises a set of bio-inspired methodologies, whose main goal is the global optimization of functions. They can also be viewed as searching algorithms, suitable in situations where the search space is large, because they explore a space using heuristics inspired by nature. Any optimization problem has to be represented by an encoded representation of the variables in the space problem. In this space, a population of candidate solutions is evaluated by a fitness function in terms of the performance. The best candidates evolve and pass some of their characteristics to their offsprings. The Genetic algorithms [51], particle swarm optimization [30] and the ant colony optimization [28], are example algorithms of the evolutionary computation.

In general, the use of such methodologies is confined to the optimization of the parameters of a segmentation method, by using a given measure of performance as a fitness function. Examples of their application to image segmentation are surveyed in [100].

1.5 Concluding remarks of the chapter

Image segmentation is one of the fundamental problems in computer vision. Its importance is based on its use as a pre-analysis of images in the development of high-level tasks, where the performance largely depends on the accuracy and robustness of the

segmentation step. Segmentation refers to the process of partitioning a digital image into separate and homogeneous regions, using a feature homogeneity criterion and a spatial connection norm.

A considerable number of approaches for image segmentation have been proposed. Such methods have been categorized according to the human interaction, the type of images they are used for and the number and type of features used in the homogeneity criterion. In this chapter, we focused the literature review according to the features, where the color, texture and the combination of these two are highlighted. It can be remarked that the combination of color and texture features in the segmentation process has shown to be strongly linked to the human perception.

Although image segmentation is one of the fundamental problems within computer vision research, it still represents a challenging task due to the difficulty in the management of different uncertainties in the images. In this regard, the use of computational intelligence, specifically the use of rough sets is proposed in this thesis. Such techniques, in contrast to classical hard computing techniques, are tolerant to imprecision, uncertainty and partial truth.

In the following chapter, special attention is given to the formal definition of rough sets in a general context and then, to their use in image analysis. Furthermore, the specific use of rough set concepts for image segmentation tasks is also presented.

CHAPTER 2

Rough sets for image segmentation

The inherent presence of imprecision, uncertainty and partial truth when dealing with image processing and analysis tasks is the reason of the growing interest for the use of rough set-based techniques. Concepts present in image processing, e.g. regions, edges, shapes, are not always precisely defined. Hence, any decision made at a particular processing level, will have an impact on all higher-level activities. Therefore, rough set-based methods have been proposed for different image processing and analysis tasks, including classification, enhancing and segmentation. More details about rough sets' definition, their background theory and implementation for image analysis and segmentation tasks are given in the following sections.

2.1 Rough sets definition

Rough set theory offers one of the most-recent approaches for modeling imperfect knowledge. This theory has been proposed by Z. Pawlak [101] as an alternative to fuzzy sets' theory and tolerance theory. In [101], Pawlak proposes approximate operations on sets,

approximate equality of sets, and approximate inclusions of sets. Rough sets have many important advantages for data mining, such as providing efficient algorithms for finding hidden patterns in data, finding minimal sets of data, generating sets of decision rules, and they offer a straightforward interpretation of results. The data can be acquired from measurements or from human experts [62]. In the last two decades, rough sets have been widely applied to data mining and rapidly established in many real-life applications such as medical diagnosis, control algorithm acquisition and process control and image processing. The main advantage of rough set theory is that it needs no a-priori knowledge or additional information about data, like, for instance, membership functions in fuzzy set theory.

Rough set theory [101] has been proposed as a mathematical tool for handling uncertainty. It offers powerful tools to extract hidden patterns from data, and therefore, it is becoming very important in various application fields. This theory includes two main parts: the knowledge description and the set approximation.

Intuitively, knowledge can be perceived as a body of information about some parts of reality, which constitutes our domain of interest [102]. Knowledge is based on the ability to classify objects, and an object it is meant to be anything we can think, i.e., real things, states, abstract concepts, processes, etc.

The basic concept for data representation in the rough set framework is an information system, represented as a table, where each row represents an object and every column represents an attribute. Formally, an information system I can be defined in terms of a pair $I = (U, A)$, where U is a non-empty finite set of objects and A is a non-empty finite set of attributes. Each attribute $a \in A$ can be viewed as a function that maps elements of U into a set V_a, $a : U \to V_a$. The set V_a is called the value set of attribute a.

A decision system (a decision table) expresses all the knowledge we have about the model we are studying. This table may be unnecessarily large because indiscernible objects may be represented several times, or some of the attributes may be superfluous.

In order to detect such issues, the notion of equivalence is defined. A binary relation $R \subseteq U^2$, which is reflexive ($xRx, \forall x \in U$), symmetric (if xRy then yRx, $\forall x, y \in U$) and transitive (if xRy and yRz then xRz, $\forall x, y, z \in U$), is called equivalence relation. The equivalence relation of an element $x \in U$ consists of all objects $y \in U$ such that xRy.

Let $I = (U, A)$ be the information system, then with any $B \subseteq A$ there is associated an equivalence relation $IND_A(B)$:

$$IND_A(B) = \{(x, x') \in U^2 | \forall a \in B a(x) = a(x')\}. \tag{2.1}$$

The equivalence relation $IND_I(B)$ is called B-indiscernibility relation. If $(x, x') \in IND_I(B)$ means that x and x' are indiscernible from each other by attributes from B. The equivalence classes of the B-indiscernibility relation are denoted by $[x]_B$.

An equivalence relation induces a partitioning of the universe. The subsets that are most often of interest have the same value of the outcome attribute. In the rough sets theory, equivalence relations are extended to classes, which are generalizations of sets. Rough sets are interested in concepts which form a partition (classification) of the universe U, in classes $C = X_1, X_2, \ldots X_n$ such that $X_i \subseteq U$, $X_i \neq \emptyset$,$X_i \cap X_j = \emptyset$ for $i \neq j$, $i = 1, 2, \ldots, n$ and $\bigcup X_i = U$.

A rough set is an approximation of a vague concept by a pair of precise concepts. Having an information system, it may happen that a specific concept cannot be defined in a precise manner. It could be possible to delineate the objects that have a given attribute with certainty, the objects that do not have such attribute and, finally, the objects that belong to a boundary between such two cases. If this boundary is non-empty, the set is rough.

If $I = (U, A)$ is an information system, let $B \subseteq A$ and $X \subseteq U$. We can approximate X using the information contained in B by constructing a $B - lower$ and $B - upper$ approximations of X. The set of all objects which can be classified with certainty as members of X with respect to B is called the B-lower approximation of a set X, and is denoted by

$$\underline{B}(X) = \{x | [x]_B \subseteq X\}. \tag{2.2}$$

The set of all objects which can be only classified as possible members of X with respect to B, is called the B-upper approximation of a set X, denoted by

$$\overline{B}(X) = \{x | [x]_B \cap X \neq \emptyset\}. \tag{2.3}$$

The set of those objects that cannot be decisively classified into X on the basis of knowledge in B is called the boundary region, and is denoted by

$$BN_B(X) = \overline{B}(X) - \underline{B}(X) \tag{2.4}$$

A set X is called crisp (exact) with respect to B if and only if the boundary region $BN_B(X)$ is empty. A set X is called rough (inexact) with respect to R if and only if the boundary region of X is nonempty. The lower approximation of a set is the union of all granules which are entirely included in the set. The upper approximation of a set is the union of all granules which have non-empty intersection with the set. The boundary region of a set is the difference between the upper and the lower approximations. The composition of a rough set is illustrated in Fig. 2.1.

In [101], Pawlak discusses two numerical characterizations of imprecision for a subset X in the approximation space $\langle U, B\rangle$: accuracy and roughness. The accuracy of X, which is denoted by $\alpha_B(X)$, is the ratio of the number of objects in its lower approximation to that in its upper approximation, namely:

$$\alpha_B(X) = \frac{|\underline{B}(X)|}{|\overline{B}(X)|} \tag{2.5}$$

The roughness of X, which is denoted by $\rho_B(X)$, is defined as $\rho_B(X) = 1 - \alpha_B(X)$. Note that the lower the roughness of a subset, the better is its approximation. Furthermore, the following properties are noted:

1. $0 \leq \rho_B(X) \leq 1$

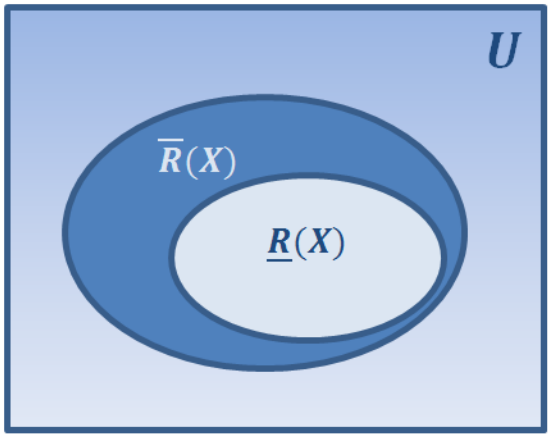


Figure 2.1: Illustration of a rough set composition.

2. $X = \emptyset \rightarrow \underline{B}(X) = \overline{B}(X) = \emptyset \rightarrow \rho_B(X) = 0$

3. $\rho_B(X) = 0$ iff X is definable in $\langle U, B \rangle$

2.2 Rough sets in image analysis

As it has been mentioned, rough sets provide a wide spectrum of practical solutions for image analysis problems such as image understanding, image classification, pattern recognition, image retrieval, perceptual relations of images and image segmentation.

The pioneers in the use of rough sets for image analysis were A. Mrosek and L. Plonka [90]. They stated that the essence of rough set-based approaches for image analysis tasks consists in viewing an image as a universe of a certain information system as a set of points (pixels). The features of the points are the source of knowledge. The earliest approach in image analysis with rough sets is the work by Skowron [120] and Bloch [5] for the approximation of mathematical morphology.

After the initial works, different authors have employed the rough sets for a variety of image analysis tasks. Rough set frameworks have been hybridized with other computational intelligence technologies that include neural networks, particle swarm optimization, support vector machines, and fuzzy sets. The most popular usage of

rough sets has been in medical imaging [50]. However, they have been applied in solving different problems such as object extraction [99], image classification [49, 144, 125], image retrieval [39] and image segmentation [116, 81].

Although image segmentation is one of the fundamental problems within computer vision research, it still represents a challenging task due to the difficulty in the management of different uncertainties in the images. As it has been mentioned in the literature review, among the spatially-blind techniques, the histogram-based methods offer advantages over other spatially-blind and spatially-guided segmentation techniques. An example of those advantages is that in the histogram-based methods, it is not required a-priori information about the image, like the number of classes or their initialization. However, the disadvantages of this approximation include the sensitivity to noise and intensity variations, the difficulty to identify significant peaks in the histogram, and the lack of representation of the spatial relationship between neighbor pixels.

As mentioned before, the lower approximation is a description of the universe of objects that are included in the set with certainty, whereas the upper approximation is the definition of the objects that possibly belong to the set. From this concept of a rough set and in the context of image segmentation with histogram-based methods, Mohabey and Ray [89] have developed the idea of the histon which can be considered as the upper approximation of a rough set and the regular histogram is considered as the lower approximation. The histon is an enhancement of the histogram, and it is a representation that associates pixels which are similar in features and may belong to one specific object in the image. The histon has the advantage of associating pixels with alike colors within a spatial neighborhood, resulting in a method which is tolerant of small variations of colors and noise. Additionally, the histon facilitates the selection of significant peaks because they are heightened in comparison with the peaks in a histogram.

2.3 Rough set-based image segmentation

In order to set the histon definition in the context of a histogram-based segmentation method, let $I(m, n, C)$ be the pixel (m, n) of an $M \times N$ image I, where m, n are the image coordinates $m \in [0, M-1]$ and $n \in [0, N-1]$. The parameter C denotes the feature channels $C = \{c_1, c_2, ..., c_j\}$, with $0 < i \leq j$ of them used in the image representation. In this study we have three information channels $j = 3$, each channel having L_i possible values. Therefore, $I(m, n, c_i) \in [0, L_i - 1]$ is the pixel value for the component i of the image at the coordinates (m, n).

The histogram of an image I is a representation of the frequency distribution of all the intensities that occur in the image. The histogram of a given color channel i is computed as in Eq. 2.6.

$$h_i(g) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \delta(I(m, n, c_i) - g) \tag{2.6}$$

where $\delta(\cdot)$ is the Dirac impulse and g is a given value $0 \leq g \leq L_i - 1$.

As previously mentioned, the histogram-based segmentation methods identify each object in the image by a peak in the histogram, making the assumption that the features on the surface of the objects are unvarying. Unfortunately, such an assumption is not always true and variations in the features are commonly found, making the identification of peaks a challenging task. Toward the solution of these uncertainties, the histon associates pixels that are similar, and possibly belong to one specific object in the image. Such association is not limited to feature similarity, it also includes the spatial relationship of the pixels and their neighbors.

Regarding the histon definition as the upper approximation of a rough set, let us consider the similarity between a reference pixel and its neighbors be the weighted Euclidean distance $d(m, n)$ defined in Eq. 2.7.

$$d(m,n) = \sum_{p,q \in W} \sqrt{\sum_{i \in C} \omega_i (I(m,n,c_i) - I(p,q,c_i))^2} \tag{2.7}$$

where $I(m,n,c_i)$ is the value of the reference pixel, $I(p,q,c_i)$ is the value of the (p,q) neighbor within the analysis window W and ω_i is a weight added to tune the contribution of each information channel.

The pixels that are similar to their neighbors are registered in an $X(\cdot)$ matrix (Eq. 2.8), where the similarity threshold is defined by a parameter named expanse and denoted by Ex,

$$X(m,n) = \begin{cases} 1, & d(m,n) < Ex. \\ 0, & \text{otherwise.} \end{cases} \tag{2.8}$$

The histon H_i, where i is a given component in the representation space, is computed as in Eq. 2.9,

$$H_i(g) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} (1 + X(m,n)) \delta(I(m,n,c_i) - g), \tag{2.9}$$

where $\delta(\cdot)$ is the Dirac impulse and g is the pixel value, where $0 \leq g \leq L_i - 1$.

The histon, in analogy to the histogram, records the frequency of occurrence of pixels that are similar to its neighbors. For each pixel that is related in features to its neighbors, the corresponding bin g, in its channel i, is incremented twice. This double increment in the histon results in the heightening of peaks, corresponding to locally uniform intensities. The main advantage of using the histon instead of the regular histogram is that the histon can capture the local similarity, resulting in a representation tolerant of small variations, and furthermore, since the peaks are heightened, their detection is easier.

In the rough-set theory, the lower and upper sets may be correlated using the concept of roughness index. A further improvement to the histon, is the proposal by Mushrif and Ray [91] (from now on referred to as RBM, for Roughness-Based Method), where

both the histon and the histogram are correlated into a roughness index representation. The roughness index is the granularity, or accuracy of an approximation set. In our scope, the roughness index, for each g value, is defined as shown in Eq. 2.10.

$$\rho_i(g) = 1 - \frac{|h_i(g)|}{|H_i(g)|}, \tag{2.10}$$

where i is the feature channel, $h(\cdot)$ is the regular histogram, $H(\cdot)$ is the histon. The value of roughness is close to 1 if the number of registers of a given bin in the histon is large in comparison to the registers of the same bin in the histogram. This situation occurs if the features on a certain region are homogeneous. The roughness tends to be close to 0 if there is a small similarity and a high variability in the region, because the histon and the histogram have the same values. The dilatation or contraction of such a boundary region between the histon and the histogram are influenced by the parameters W and Ex. When the similarity tolerance Ex is too big, the association of regions becomes more flexible, possibly leading to under-segmentation. When Ex is small, the association becomes more rigid, making possible that even pixels with a high similarity will be considered as different regions, resulting in an over-segmentation.

To achieve good segmentation results using rough set-based methods, the selection of peaks and thresholds from the roughness index is very important. In the methods proposed by Mushrif [91] [92], the selection of peaks is carried out on the roughness index for each color component in the RGB color space. The criteria used for the selection of the significant peaks is based on two rules, empirically determined. The first criterion establishes a specific height, and the other defines a minimum distance between two peaks. The height of a significant peak must be above the 20% of the average value of the roughness index for all the pixel intensities, and the distance between two significant peaks has to be higher than 10 units.

Although the roughness index-based method provides interesting segmentation results, one has to be aware that although those fixed criteria for the threshold selection are easy to follow, they might not be appropriate for different images. Therefore, since

the thresholds are dependent on the image content, it is desirable to use adaptive criteria, in the selection of the optimal thresholds.

2.4 Concluding remarks of the chapter

In this chapter, the definition of rough sets and the use of this concept for the image segmentation problem has been presented. Special attention has been established to the roughness index-based segmentation approach named as RBM. This methodology provides some advantages over other approximations, since it does not require a priori information about the image under analysis, like the number of classes or their initialization. Moreover, this method has the advantage of associating pixels with similar colors within a spatial neighborhood, resulting in a method which is tolerant of small variations of colors and noise. However, as it has been pointed out, the selection of thresholds might not be adequate for different images, and adaptive criteria must be used.

In the next chapter, a set of improvements to the algorithm proposed by Mushrif [91] are introduced, and this new approach is evaluated in comparison with other related techniques and standard segmentation methods.

CHAPTER 3

Improving a rough set-based segmentation approach for color images

In this chapter, we propose a color image segmentation approach based on the rough set theory. The contribution of the proposed approach in comparison with the previous rough set-based approaches is threefold. First, a study is accomplished in order to determine the best-suited color representation for the segmentation approach. Second, with an adaptive threshold selection, the approach is automatically adjustable according to the image content. Third, concerning over-segmentation, the use of a region merging process, which takes into account the features and the spatial relations of the resulting segments, is proposed. These three strategies help our method to overcome some performance issues shown by previous rough set theory-based approaches.

This chapter is divided in three sections. In Section 3.1, the modifications and adaptations performed in order to improve the rough set-based segmentation approach are described. In Section 3.2, the experiments and results are given, followed by concluding remarks in Section 3.3.

3.1 Improving a rough set theory based segmentation approach

The method proposed in this chapter lead to the segmentation approach illustrated in Fig. 3.1. From now on, this method is referred as PRM, for Perceptual Roughness index-based Method.

At the beginning of the PRM, a color space transformation is applied to the input image. Then, the rough set theory-based approach with a multithreshold method and an adaptive peak selection is performed on each component of the transformed image. The main improvement is the adaptive peak selection, where the criteria used for choosing the significant peaks change according to the image content. At the end, the segmented image is obtained after a region merging process, which takes into account both feature similarity and spatial relationship. Each block of the Fig. 3.1 is described in the following subsections.

3.1.1 Definition of the color spaces under analysis and color space transformations

A color space is an abstract representation which describes colors as tuples of numbers, typically as three values or color components [31]. A color space is a coordinate system, where each color is represented by one point in the space.

In the context of image segmentation, it is known that the use of a given color representation has a relevant impact on the performance of a segmentation method [6]. In this regard, we explore the use of different color spaces and their impact on the performance of our method. We have mainly explored the use of the perceptual color spaces ($CIELab$ and $CIELuv$) and the RGB space. In this subsection, we describe the required transformations to take an image from the RGB color space to the perceptual color spaces explored in this study.

The RGB color space is an additive color space based on the RGB color model. The

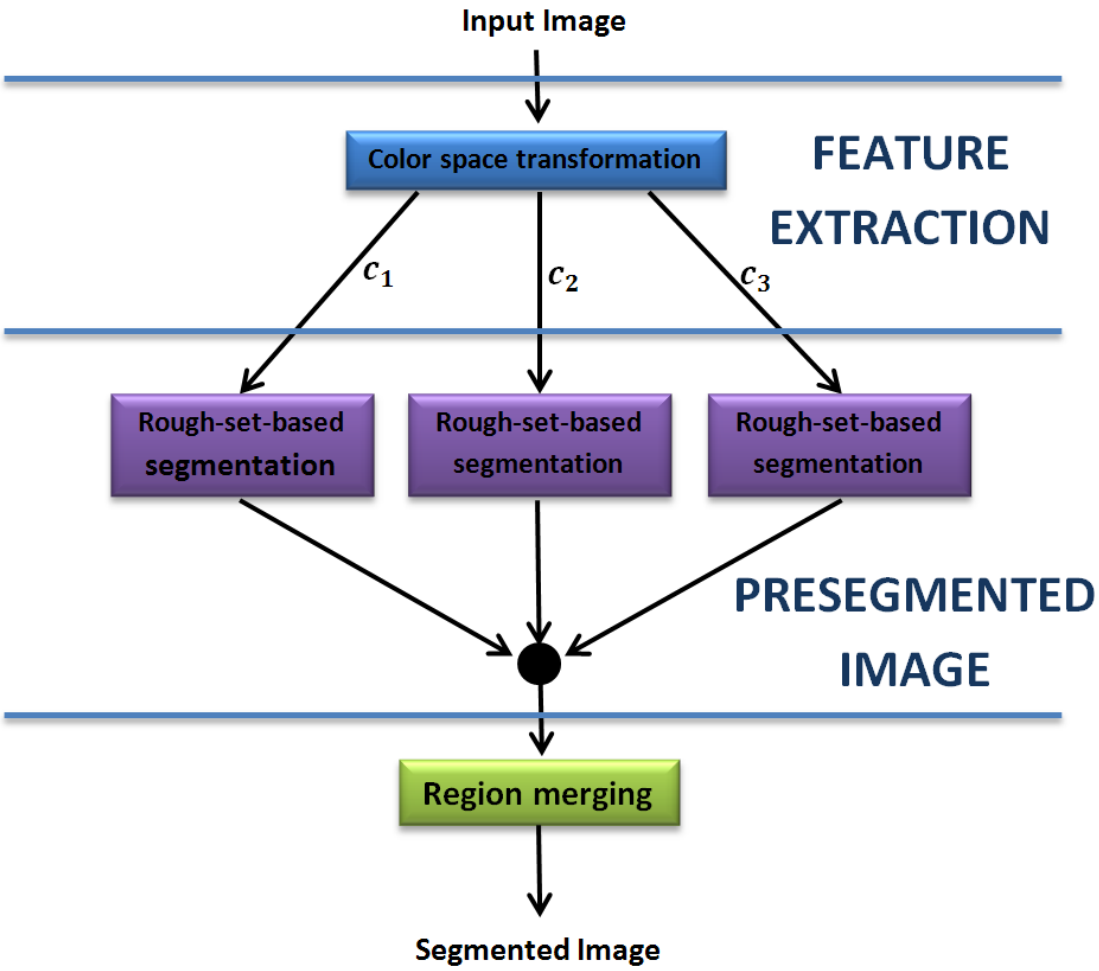


Figure 3.1: The general process of the proposed segmentation approach.

main purpose of the RGB space is the sensing, representation and display of images in electronic systems, such as televisions and computers, though it has also been widely used in conventional photography. A particular RGB color space is defined by the three chromaticities of the red, green, and blue additive primaries. In this space, each color component can take values in a normalized range $[0, 1]$ or in a discrete range $[0, L-1]$, where $L = 2^{nb}$ and nb represents the number of bits used in the discretization process. Usually, $nb = 8$ and therefore, there are $L = 256$ possible values in each color component.

Given that the perceptual color spaces, $CIELab$ and $CIELuv$, transformations are applied to the CIEXYZ space, the transformation from RGB to CIEXYZ space must be reviewed. The color representation CIE 1931 XYZ, best known as CIEXYZ, is one of the first color spaces obtained from a mathematical model of the human color perception. It was developed by the Commission Internationale de l'Èclairage (CIE), back in 1931 [121]. The transformation equation is presented in Eq. 3.1,

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124564 & 0.3575761 & 0.1804375 \\ 0.2126729 & 0.7151522 & 0.0721750 \\ 0.0193339 & 0.1191920 & 0.9503041 \end{bmatrix} \begin{bmatrix} r \\ g \\ b \end{bmatrix}, \tag{3.1}$$

where $rgb \in [0, 1]$ are the normalized values computed by $\{r, g, b\} = \frac{1}{L-1}\{R, G, B\}$, for an image with L possible values.

The CIE 1976 (L^*, a^*, b^*) color space, better known as $CIELab$, is a space derived from the CIEXYZ color space. The transformation equations to obtain the Lab components are defined from Eq. 3.2 to Eq. 3.5,

$$f(t) = \begin{cases} t^{1/3}, & t > \alpha^3, \\ t/(3\alpha^2) + 16/116 & t \leq \alpha^3. \end{cases} \tag{3.2}$$

$$L = 116 f(Y/Y_n) - 16, \tag{3.3}$$

$$a = 500[f(X/X_n) - f(Y/Y_n)], \tag{3.4}$$

$$b = 200[f(Y/Y_n) - f(Z/Z_n)], \tag{3.5}$$

where $\alpha = 6/29$, and (X_n, Y_n, Z_n) is the white point for the scene in CIEXYZ. In this work we have used the standard for a daylight illuminant D65, where $X_n = 0.9504$, $Y_n = 1.000$ and $Z_n = 1.0887$.

In the same manner as the $CIELab$ color space, the $CIELuv$ was developed by the CIE in order to obtain a perceptually linear space. The transformation equations from the CIEXYZ space to $CIELuv$ are defined in Eqs. 3.6 to 3.10.

$$u' = \frac{4X}{X + 15Y + 3Z}, \tag{3.6}$$

$$v' = \frac{9Y}{X + 15Y + 3Z}, \tag{3.7}$$

$$L = 116(Y/Y_n)^{1/3} - 16, \tag{3.8}$$

$$u = 13L(u' - u'_n), \tag{3.9}$$

$$v = 13L(v' - v'_n), \tag{3.10}$$

where Y_n is the Y component of the reference white in CIEXYZ and u'_n and v'_n are the chromaticity coordinates of the reference white. The inverse transformations of these color spaces are not required because in our study, there is no need to return to the RGB color representation.

The adaptation of the PRM procedure to use perceptual color spaces is simple. The input image is in the RGB color space with $L_i = 256$ possible values in each channel i, $0 \leq g \leq 255$. Once the whole image has been transformed to a given color space, each component is fitted to the range of $[0, 255]$ by a linear dynamic range expansion of the image.

3.1.2 Roughness index-based segmentation

Posterior to the transformation and adjustment of the color space, the second issue to address is the peak and threshold selection procedure in the roughness index rep-

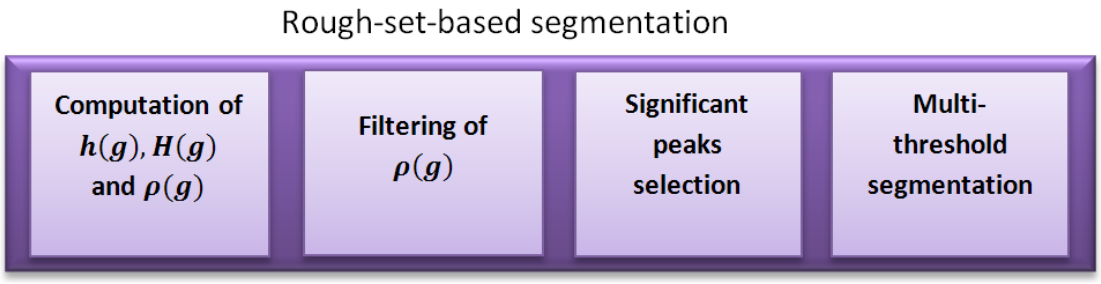


Figure 3.2: Internal steps of the proposed rough-set-based segmentation.

resentation. In the Mushrif and Ray method [91] (referred to as RBM), the selection of peaks is accomplished by following a succession of fixed criteria. Therefore, there is no guarantee that the best peaks are found for all types of images. In this regard, the selection of the thresholds must be adaptive, in such a way that different height and distance criteria are used for distinct images.

The main steps of the rough-set-based segmentation approach are shown in Fig. 3.2. First, the histogram $h(g)$, the histon $H(g)$ and the roughness index array $\rho(g)$ are computed of each image channel c_i. After that, the array $\rho(g)$ is filtered in order to diminish small variations. The adaptive selection of the significant peaks in the roughness index is performed and, finally, a multithreshold segmentation is accomplished. More details are given in the following paragraphs.

Concerning the reduction of small variations in the roughness index array $\rho(g)$, we have tested both linear and non-linear filters such as averaging filters and rank filters with different window sizes. We have found that the linear filter shown in Eq. 3.11 offers the best results in our segmentation process.

$$\rho_i'(g) = \frac{\rho_i(g-2) + \rho_i(g-1) + \rho_i(g) + \rho_i(g+1) + \rho_i(g+2)}{5}. \tag{3.11}$$

In the histogram-based methods, the valley points represent the object boundaries and the region between the two valley points denotes the object region. Therefore, in the third step of our method, we determine the relevant peaks and valleys of the

roughness index, and get the feature values of the objects in the image. The selection of the significant peaks is important for achieving good segmentation results. The criteria used for their selection are based on the height of the peaks and the distance between two significant peaks. The criteria are: (a) The peak is significant if the height of the peak is greater than T_h and (b) the peak is significant if its distance to the previous peak is greater than T_d.

The computation of the criteria thresholds, T_h and T_d, is accomplished adaptively for each image and for each of its channels. First, the set of all local maxima $P = \{p_1, p_2, \ldots, p_j, \ldots, p_k\}$ is obtained from the filtered roughness index $\rho'(g)$. The roughness index is considered to have a local maximum p_j on g if $\rho'(g) \geq \rho'(g-1)$ and $\rho'(g) \geq \rho'(g+1)\}$. After the identification of the k local maxima, the mean and the standard deviation of their corresponding heights (μ_h, σ_h) and distances (μ_d, σ_d) are computed as in Eqs. 3.12 and 3.13.

$$\mu_h = \frac{1}{k}\sum_{j=1}^{k}\rho'(p_i), \qquad \sigma_h = \sqrt{\frac{1}{k}\sum_{j=1}^{k}[\rho'(p_j) - \mu_h]^2} \tag{3.12}$$

$$\mu_d = \frac{1}{k-1}\sum_{j=2}^{k}(p_j - p_{j-1}), \qquad \sigma_d = \sqrt{\frac{1}{k-1}\sum_{j=2}^{k}\left[(p_j - p_{j-1}) - \mu_d\right]^2} \tag{3.13}$$

The set of n significant peaks contains the intensity levels g whose roughness value is above the height threshold $T_h = \mu_h - \sigma_h$ and that also have a distance to the previous peak greater than $T_d = \mu_d - \sigma_d$.

After the set of n significant peaks is computed, the multithresholding process is performed. The thresholds $T = \{T_1, T_2, \ldots, T_{n-1}\}$ are localized on the valleys (the minima values) found between two significant peaks. The pre-segmented component S_i is obtained by assigning a label to every pixel with intensity under a threshold T (See Eq. 3.14). In Fig. 3.1.2 the process to determine the best thresholds is summarized. After the roughness index is filtered (Fig. 3.1.2(a)), the set of all local maxima

are obtained (Fig. 3.1.2(b)). The significant peaks are selected by using the criteria established (Fig. 3.1.2(c)) and the thresholds are found on the minimum value between two significant peaks (Fig. 3.1.2(d)).

$$S_i(m,n) = \begin{cases} l_1, & I(m,n,c_i) < T_1, \\ l_2, & T_1 \geq I(m,n,c_i) < T_2, \\ \vdots & \\ l_{n-1}, & T_{n-2} \geq I(m,n,c_i) < T_{n-1}, \\ l_n, & I(m,n,c_i) \geq T_{n-1}. \end{cases} \tag{3.14}$$

When all the pre-segmented components $S_i, i = \{L, a, b\}$ are computed, we obtain the final pre-segmented image S by finding the union of the three components $S = S_L \cup S_a \cup S_b$. In Fig. 3.1.2, an example of the pre-segmented image obtention is presented.

3.1.3 Region merging

The final step in the PRM segmentation framework is the reduction of possible over-segmentation issues with a region merging process. The image obtained with the union of the three pre-segmented image components often results into an over-segmented image. Therefore, the final segmented image is obtained by applying a region merging algorithm. This process fuses small regions with the most appropriate neighbor segments. The fusion criteria for the merging step vary from method to method. Usually, the region merging is based on both, features and the spatial relation between pixels simultaneously. Nonetheless, some methods [91, 92, 15] only consider feature similarity to decide if two regions are to be fused, ignoring the spatial relationship of the different segments.

In our method, the region merging step takes into account both, color similarity and spatial relationship between regions. The strategy, firstly identifies the small regions whose number of pixels is less than a given threshold. From the experiments, we have found that a good minimum number of pixels in a given region to be considered as small

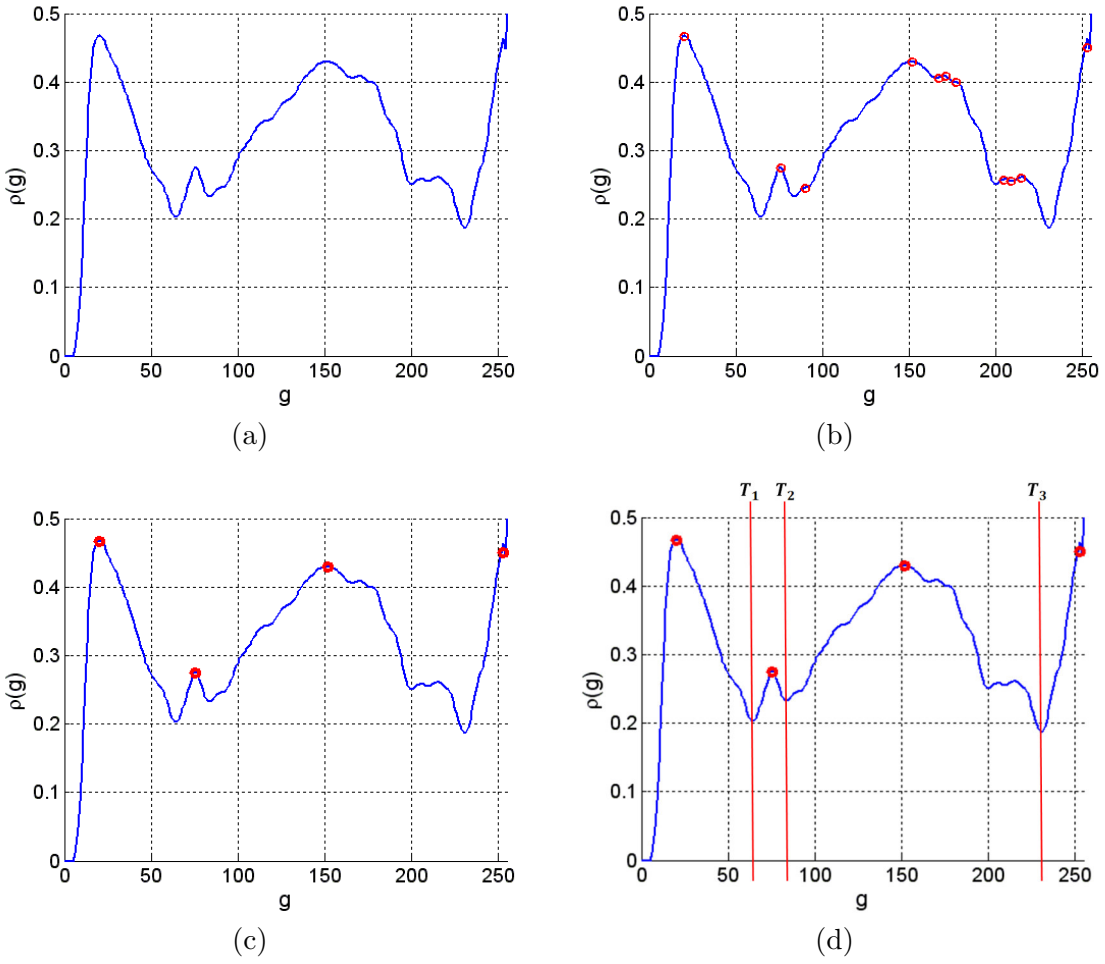
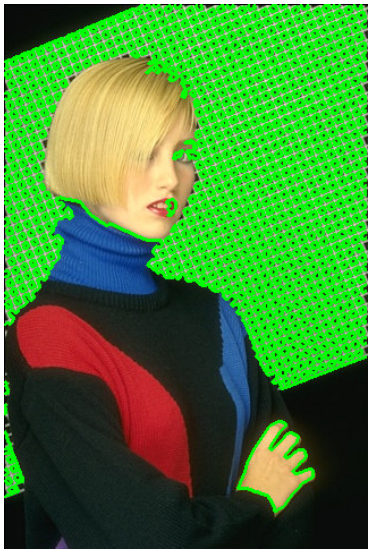
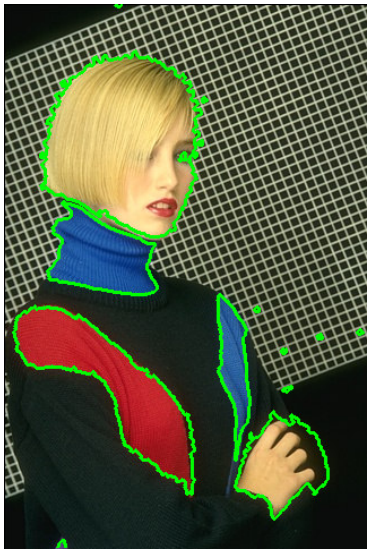


Figure 3.3: Process to determine the best thresholds for an image. (a) Filter the roughness index. (b) Obtain the set of all the local maxima in the array. (c) Select the significant peaks by using the criteria established. (d) Set the thresholds on the minimum value between two significant peaks.

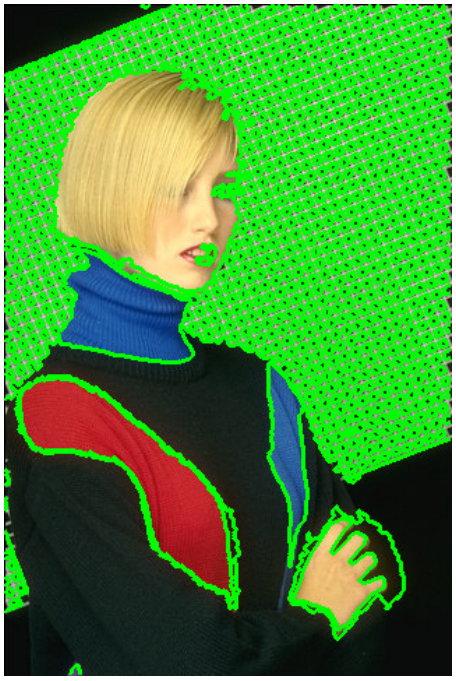


(a) S_L

(b) S_a



(c) S_b



(d) $S = S_L \cup S_a \cup S_b$

Figure 3.4: An example of the segmented image channels and their union into the pre-segmented image.

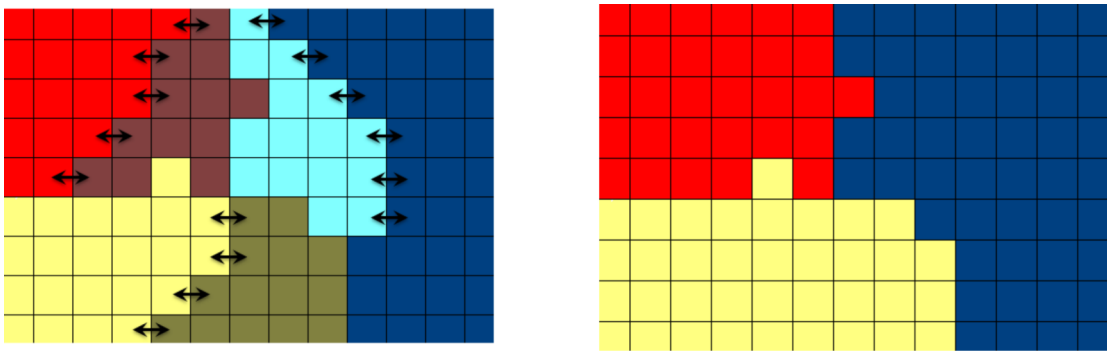


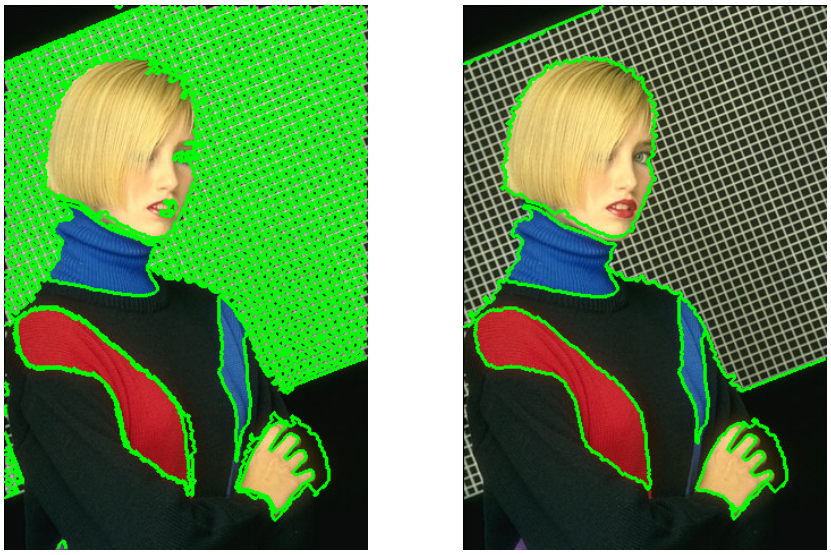
Figure 3.5: Initial regions (left) and the final segmentation map after the region merging process (right) using the two criteria of feature similarity and spatial connectivity.

is 0.2% of the whole image size. Once we have identified all the small regions, they are fused with the most appropriate neighbor region. Such region is the one that minimizes the distance between the mean values of the regions and maximizes the number of connected pixels between those two regions. In our approach, we have used the 4-connectivity. Hence, a small region is merged with the neighbor who is more similar in features and has more connected pixels. In case of conflict, the feature similarity is privileged. The process is illustrated in Fig. 3.5.

In Fig. 3.6, an example of the merging process effect is presented. In Fig. 3.6(a) the result of the rough set-based segmentation it is shown, before the submission to the merging process. In this image, the borders of the detected segments are overlaid to the original image and we can observe that many small regions are present. The Fig. 3.6(b) shows the resulting image after the merging process, where the over-segmentation issues are significantly reduced.

3.2 Experiments and Results

In this section, we present experiments on natural scene images, in order to evaluate: (1) The performance of our proposal using different color spaces and (2) the performance



(a) (b)

Figure 3.6: Resulting images (a) before and (b) after the region merging process.

of the proposed method in comparison with other state-of-the-art approaches.

In these experiments, the Berkeley Segmentation Data Set and Benchmarks [83] is used. As it was established in Chapter 1, this data set is a well-known empirical basis for the evaluation of segmentation algorithms. The performance of our segmentation method is quantitatively evaluated adopting the three metrics introduced in Chapter 1: the Probabilistic Rand-Index (PRI) [132], the Global Consistency Error (GCE) [83] and the Boundary Displacement Error (BDE) [34].

3.2.1 Parameter tuning

Before presenting the comparison in different color spaces, let us remark that the rough-set-based segmentation methods depend on two parameters (W, Ex), the size of the window W and the expanse Ex. Ex is the similarity tolerance threshold and W is the size of the neighborhood analyzed. Hence, it is important to estimate the best

parameters to be used for each color space. In the previously proposed RBM, that uses the RGB color space, the best couple of parameters proposed by the authors is (3,100). By contrast, as the perceptual color spaces have different shapes in comparison with the RGB color space, we must estimate the best parameters for them.

The combination of parameters is evaluated using the 300 images taken from the BSD and quantitatively assessed with the PRI measure. We have exhaustively searched for the best window size W in the set $\{3, 5, 7, 9, 11\}$ and the expanse Ex in the set of $\{50, 100, 150, \ldots, 400\}$. We have found that the best parameters for the RGB space using our method are $(5, 150)$. In the case of the $CIELab$ space the best set is $(7, 150)$. Using the $CIELuv$ color space, the best parameters found are $(5, 300)$.

3.2.2 Performance evaluation in different color spaces

A quantitative comparison of the use of different color spaces in the PRM is accomplished. The corresponding PRM_{RGB}, PRM_{Lab} and PRM_{Luv} are implementations of our PRM that employs RGB, $CIELab$ and $CIELuv$ color spaces respectively, using their best parameter set. In Figs. 3.7, 3.8 and 3.9, the plots corresponding to the PRI, GCE and BDE measures are provided for each test image. The PRI values are plotted in increasing order, while for the GCE and BDE, the values are plotted in decreasing order. Hence, a given index in the horizontal axis may not represent the same image across the algorithms. In this way we can evaluate the quality tendencies of each method and compare them. As we can see from these figures, the curves related to the use of perceptual color spaces are consistently better than the curve representing the RGB space. This means that the PRM_{Lab} and PRM_{Luv} curves are over the PRM_{RGB} curve for the PRI and below it for the GCE and BDE measures. It is observed that both PRM_{Lab} and PRM_{Luv} curves are very close from each other and from a visual inspection, it is hard to determine which space is the most appropriate for our method. In this case, the mean results aim to a much clearer distinction in this evaluation.

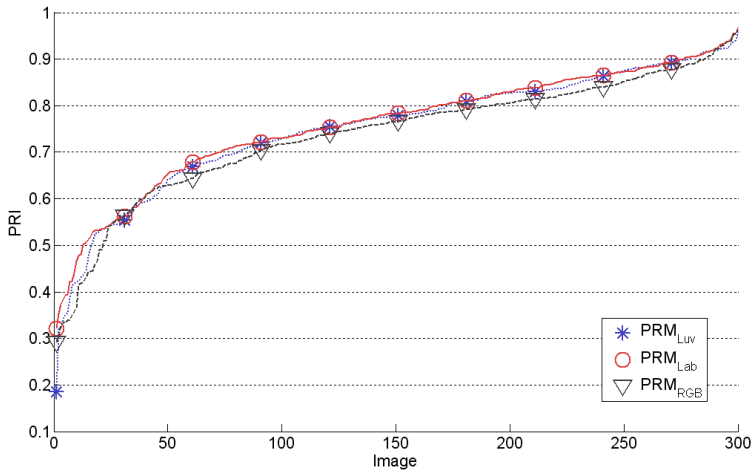


Figure 3.7: Comparison of the PRI results of our approach in the three color spaces RGB, $CIELab$ and $CIELuv$. A higher PRI value is desirable.

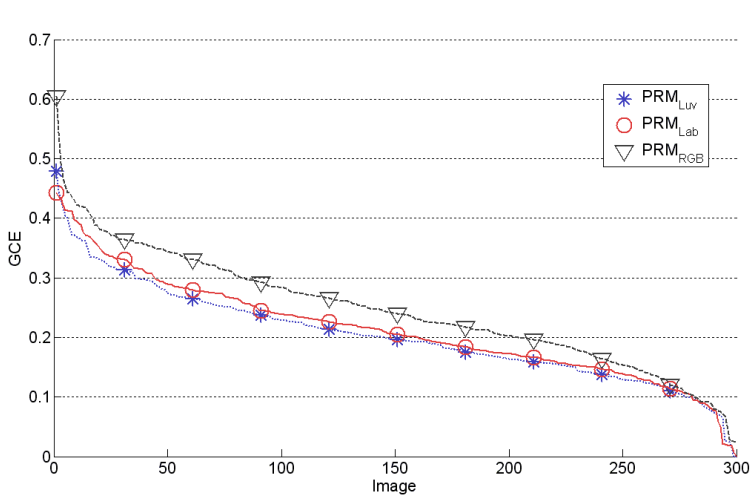


Figure 3.8: Comparison of the GCE results of our approach in the three color spaces RGB, $CIELab$ and $CIELuv$. A low GCE value is desirable.

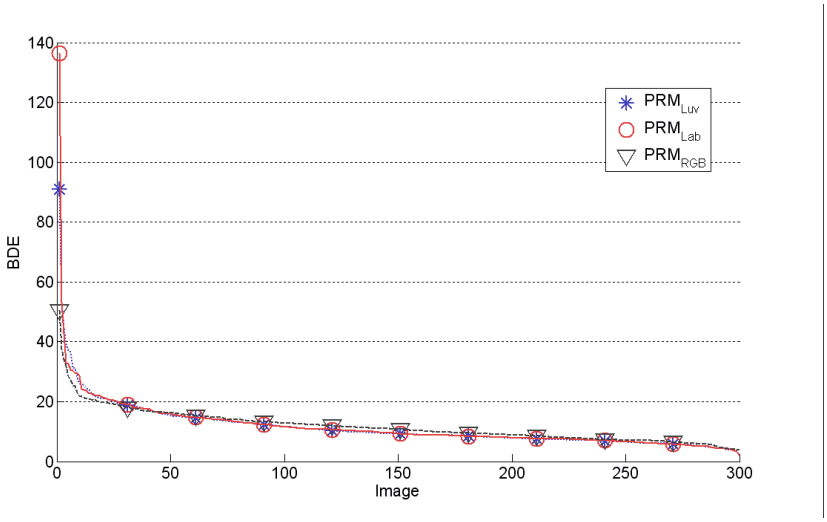


Figure 3.9: Comparison of the BDE results of our approach in the three color spaces RGB, $CIELab$ and $CIELuv$. A low BDE value is desirable.

The average PRI, GCE and BDE results for each color space with the 300 images are detailed in Table 3.1, where the best result for each measure is highlighted with a bold typeface. In this table, the higher PRI average is achieved by PRM_{Lab} with a value of 0.760. The use of the $CIELuv$ space is slightly behind with 0.752 and the PRM in RGB gets the lower value of 0.737. The standard deviation and the number of images with a PRI value higher than 0.7 are also presented. The value of 0.7 is taken because it is empirically considered that images which obtain that PRI value, or above of it, are good segmentations. In the same manner, the best results are obtained with the PRM_{Lab}, with the lowest standard deviation and the higher number of images with PRI scores above 0.7. For the GCE and BDE averages, the best results are achieved by PRM_{Luv} with 0.203 and 11.42 respectively, while the PRM_{Lab} is slightly behind with 0.212 and 11.59 in average, for these error measures.

Qualitative examples of the segmentation results using PRM are shown in Fig. 3.10. In each row, the original image and the corresponding outcome for each color space are

Table 3.1: Average performance and comparison in three color spaces using the 300 images in the BSD.

	PRM_{RGB}	PRM_{Luv}	PRM_{Lab}
PRI	0.737	0.752	**0.760**
Std. Dev PRI	0.133	0.133	**0.126**
PRI > 0.7 (# Images)	210	219	**226**
GCE	0.245	**0.203**	0.213
BDE	11.74	**11.42**	11.59

visually compared. The original image is shown in the first row, while the resulting segmentation of the PRM using the RGB space is shown in the second row. The corresponding segmentations of the PRM_{Luv} and PRM_{Lab} are shown in the third and fourth rows, respectively. From this qualitative comparison, we can observe that the resulting segmentations of the perceptual spaces outperform the results using the RGB color representation. It can be seen that, in these examples, the outputs from the PRM_{RGB} show a clear over-segmentation, and the PRM_{Luv} and PRM_{Lab} succeed in associating regions with similar colors.

3.2.3 Performance comparison with other methods

After the study of the PRM in different color spaces, a comparison of our approach against other segmentation methods based on rough set theory, has been conducted. The comparison is carried out with the Roughness index-based technique (RBM) [91] and the Roughness approach through Smoothing Local Difference (referred to as RSLD) [139]. The RSLD method is of special interest since it is a roughness-based method performed on the perceptual color space $CIELuv$. The results for the Normalized cuts method (NCuts) [118] and the Mean Shift segmentation approach [20] are also included. The NCuts and Mean Shift methods are considered in this comparison because of their influence as widely-used methods in image segmentation tasks. Besides, they are considered as *de facto* standard references for evaluation purposes. It is important to mention that

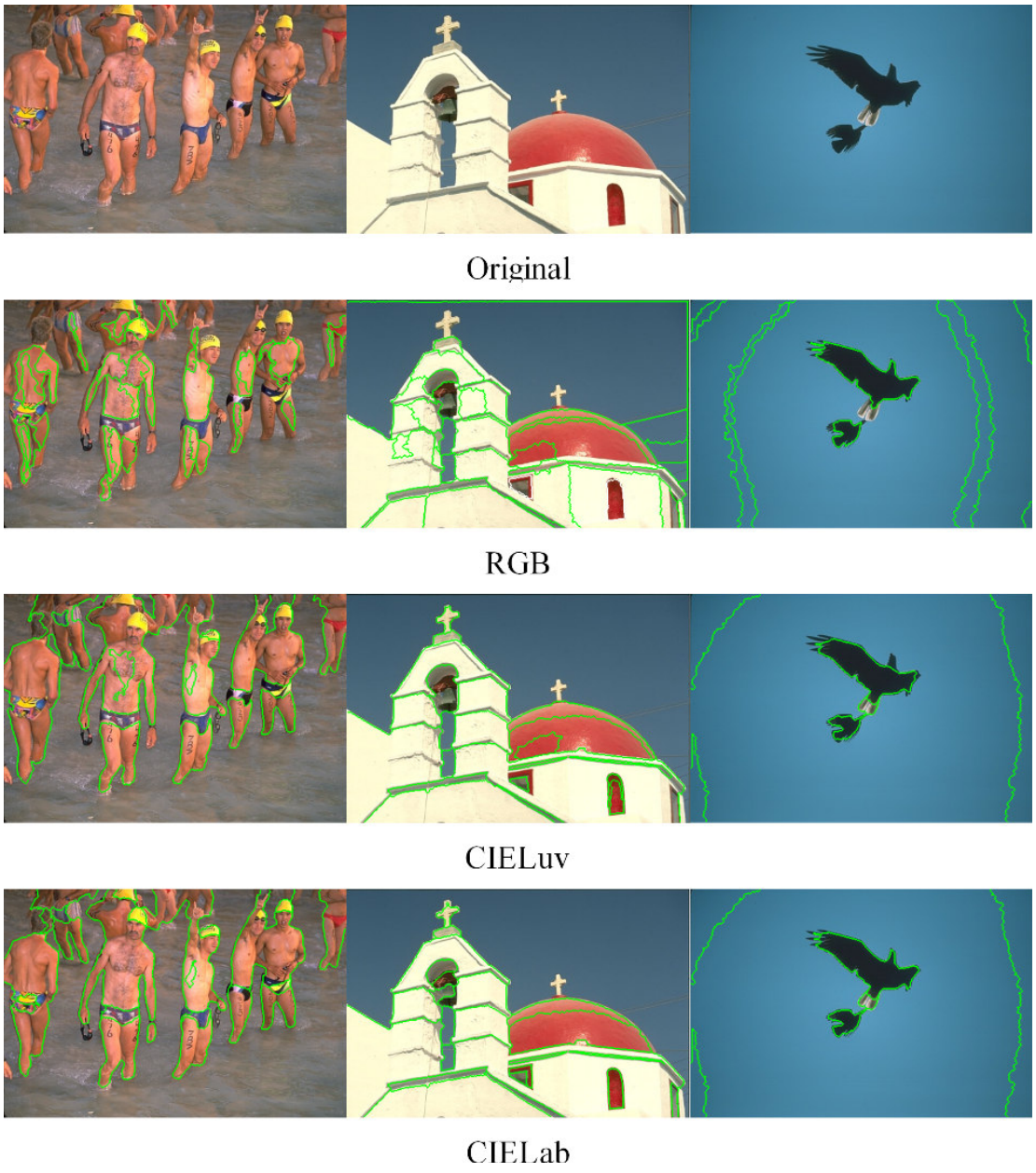


Figure 3.10: Qualitative comparison of three samples (first row) out of 300 images from the BSD in the different color spaces under analysis (RGB in the second row, $CIELuv$ and $CIELab$ in the third and fourth row, respectively).

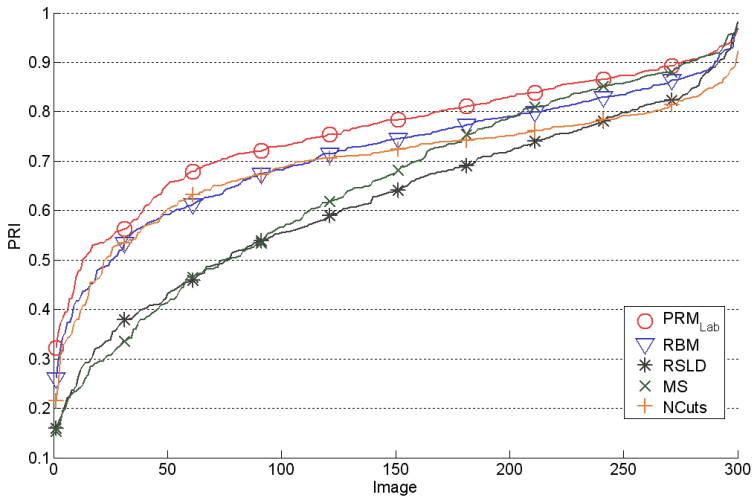


Figure 3.11: Comparison of the PRI results of our method PRM_{Lab} with other rough set-based methods (RBM and RSLD) and reference methods (NCuts and Mean Shift), using the 300 test images from the BSD. A higher PRI value is desirable.

the outcomes of the RBM, and the RSLD methods were obtained with our implementation of the algorithms described by the authors.

The corresponding results for the 300 test images in the BSD are obtained and compared. In Figs. 3.11, 3.12 and 3.13, the PRI, GCE and BDE values, respectively, are plotted. We can observe from Fig. 3.11, that the PRM_{Lab} consistently outperforms the rest of the methods. It is interesting to notice that the NCuts curve in this figure records higher PRI values than the Mean Shift, RBM and RSLD methods for the first 90 images, keeping its dominance over the Mean Shift until image number 160 and over the RSLD method until the image 240. Nevertheless, its PRI evaluation drops from there on. The reason of this trend is the low variability in the NCuts method.

In the case of Fig. 3.12, the curve with the lowest values is the one representing the Mean Shift approach, followed by our PRM. The curves with the highest errors are the curves of NCuts and RBM. It is difficult to make a visual separation of methods

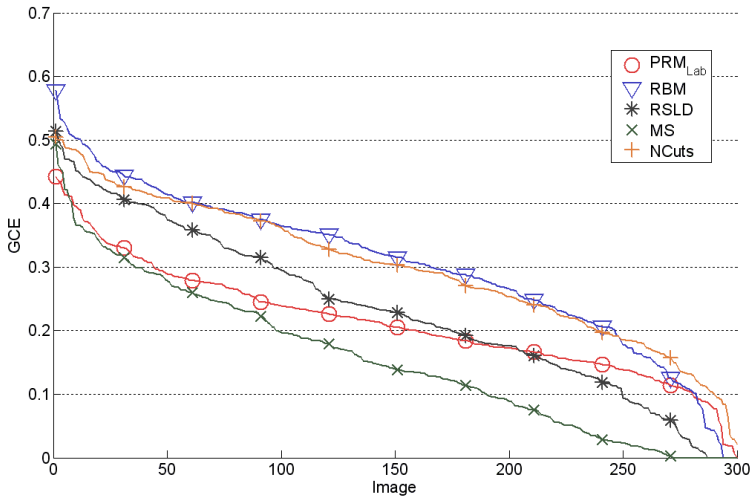


Figure 3.12: Comparison of the GCE results of the PRM_{Lab} with RBM and RSLD and the reference methods NCuts and Mean Shift using the 300 test images from the BSD. A lower GCE value is desirable.

in Fig. 3.13, since the PRM_{Lab}, NCuts, RBM and RSLD curves are very close. The corresponding curve of the Mean Shift method in the BDE measure is the only one that stands out the rest of the curves, having the highest error values.

The average PRI, GCE and BDE results for all the methods, using the 300 images from BSD, are displayed in Table 3.2. From this table, we can see that the highest average value of 0.760 is achieved by PRM_{Lab}, followed by the RBM with an average of 0.743. The lowest PRI average is obtained with the RSLD method with 0.620. In this table, again, our method obtains more images with PRI values higher than 0.7. In this case, the approximation with the lowest standard deviation, is the NCuts method with 0.119. However, our approximation is the second more precise method with a standard deviation of 0.133. In the case of GCE, the lowest error is attained by the Mean Shift method, and for the BDE, the PRM_{Lab} gets the lowest error of 11.59.

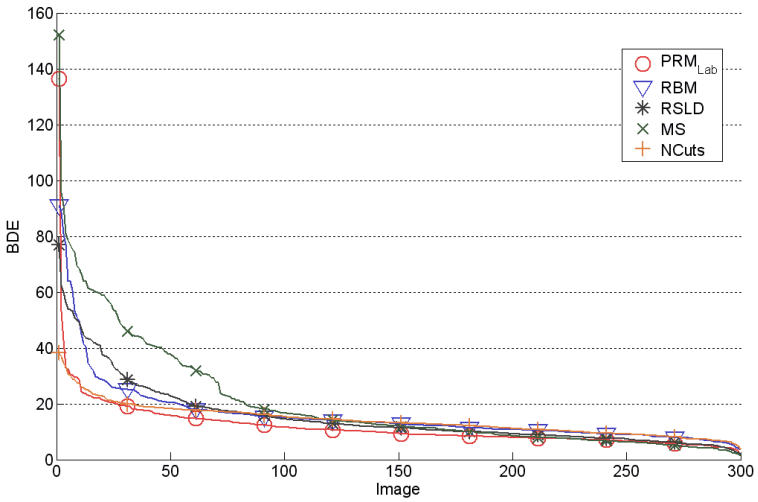


Figure 3.13: Comparison of the GCE results of the PRM_{Lab} with RBM and RSLD and the reference methods NCuts and Mean Shift using the 300 test images from the BSD. A lower GCE value is desirable.

Table 3.2: Average performance and comparison with other methods using the 300 training images in the BSD.

	PRM_{Lab}	RBM	RSLD	Mean Shift	NCuts
PRI	**0.760**	0.743	0.620	0.649	0.722
Std. Dev PRI	0.126	0.135	0.176	0.205	**0.119**
PRI > 0.7 (# Images)	**226**	180	114	160	200
GCE	0.213	0.302	0.230	**0.154**	0.298
BDE	**11.59**	15.75	15.18	19.68	14.05

3.3 Concluding remarks of the chapter

In this chapter, a set of modifications were proposed in order to improve a rough-set-based segmentation method. Such technique, in contrast to classical hard computing techniques, are tolerant to imprecision, uncertainty and partial truth, typical characteristics of natural images. The proposed improvements are basically three. First, the use of perceptually uniform color spaces instead of the RGB color representation was evaluated. We have found that the $CIELab$ color space is the most suited representation for our specific segmentation method, meanwhile the $CIELuv$ space has a performance only slightly inferior. The use of a perceptual color representation allows our system improve the association of similar colors. The second proposal is an adaptive selection of thresholds, which permits to select the most suitable thresholds for a given image. The third modification is the use of a region merging process, which includes constraints about the spatial relation and color similarity, diminishing over-segmentation problems. The results obtained by our method show that this approach has the best segmentation performance with high consistency and low error. In general, it is observed that the use of the proposed significant peaks selection, jointly with the merging strategy and the use of perceptual color spaces, improves the segmentation results with respect to the original proposal RBM. Moreover, the presented modifications allow better results in comparison with other classic segmentation methods in terms of three different quantitative measures.

According to these experiments we may highlight that the rough-set-based segmentation method has not been fully explored, especially in the use of features different than the RGB color space. In this regard, the second part of this dissertation propose the collaborative inclusion of texture features and perceptual color spaces in a rough set-based segmentation method. Previous to the presentation of this methodology, in the next chapter we present an overview of the methods for texture extraction and texture representation.

Part II

Color-Texture Image Segmentation

Introduction

In many computer vision and image processing algorithms, simplifying assumptions are made about the homogeneity of intensities in local neighborhoods within an image. However, natural images often do not show uniform intensities and exhibit visual texture, making difficult the analysis of such images. Considering that the texture is an inherent property of all objects, its study is highly relevant for any computer vision system. Many common low level algorithms, such as edge detection, break down when applied to images that contain textured surfaces. It is therefore, crucial to develop robust and efficient methods for processing images with textured regions.

In this part of the dissertation, the addition of texture features to the rough set-based segmentation approach is proposed. Hence, a comprehensive review of the texture analysis is first presented in **Chapter 4**. Additionally, the fundamental problems of texture analysis, like classification, synthesis and segmentation of texture, are reviewed. The importance of this texture analysis review is based on that texture is the second major component of the proposed segmentation framework proposed in **Chapter 5** and it is used to complement the color features. In this chapter, details about the integration of color and texture cues in the rough set-based segmentation approach, are presented.

It must be mentioned that in parallel to the image segmentation research, we have conducted intensive work in texture analysis. The work developed in this field is presented in this chapter, and it comprises:

- A texture classification approach using a fuzzy rule-based system (Section 4.3.1).
- A dynamic texture synthesis method using local spatiotemporal patterns (Section 4.3.2).
- A method to estimate the fundamental pattern of periodic and near-periodic textures with potential applications in defect detection and texture synthesis (Section 4.3.3).

CHAPTER 4

Overview of texture features extraction and texture analysis

Texture analysis is an important research area in computer vision and image processing. Visual texture is a perceived property on the surface of all objects and it is an significant reference for their characterization and discrimination. However, a successful application of texture analysis is complicated due to the irregularity of textures in the real world: they are not uniform and the visual appearance may change in terms of the orientation, scale, contrast, etc. These variations result in severe difficulties in finding an adequate texture representation.

Although the concept is intuitively clear, there is no formal definition for visual texture. Nonetheless, efforts have been made to propose definitions from different points of view:

- Tamura et al. [123] point out that we may regard texture as what constitutes a macroscopic region.

- Rosenfeld [109] states that texture structure is simply attributed to the repetitive patterns in which elements or primitives are arranged according to a "placement rule".

- Aribot [4] mentions that texture is a spatial concept that indicates, apart from color and gray level, visual homogeneity characteristics in a given neighborhood within the image.

- Sklansky [119] establishes that "a region in an image has a constant texture if a set of local statistics or other local properties of the picture function are constant, slowly varying, or approximately periodic."

These definitions involve fundamental aspects of the definition of visual texture analysis: the structural and the statistical analysis. From the structural point of view, it is widely accepted to define the texture as a conjunction of two components: i) a texture element (texel), which is the central microstructure in the image [143], and ii) a set of rules for texel placement into the field of view. The statistical approaches study the spatial distribution of gray values by estimating a set of statistics from the distributions of local features. The statistical analysis is appropriate for fine textures, while the structural may be considered for the coarse ones.

In general, visual texture has a number of qualitative attributes that have to be taken into account, in order to achieve a successful texture analysis.

- Texture is a region property. The texture of a point is undefined. Its definition must involve a spatial neighborhood.

- Texture is represented by the spatial distribution of gray levels.

- Texture in an image may be perceived at different scales.

- A region is perceived to have texture when the number of primitive elements in the region is large.

There is a number of perceived qualities, which play an important role in describing a visual texture. According to Laws [65], those qualitative characteristics may be uniformity, density, coarseness, roughness, regularity, linearity, directionality, frequency, and phase. In the same manner, Tamura *et al.* [123] describe the features that humans associate to the visual texture and besides, they are possible to approximate computationally. Such features are coarseness, contrast, directionality, line-likeness, regularity and roughness. Therefore, a texture may be considered as: coarse, fine, regular, irregular, directional, nondirectional, rough, smooth, etc.

Since the perception of texture has so many different dimensions, there is no single method of texture representation adequate for a great variety of textures [130]. It is observed that one approximation cannot be used for all existing textures. Sanchez-Yanez et al. [111] mention that any texture may contain both structural and statistical features, so it is desirable that a system may be able to deal with any situation [129].

4.1 Extraction of texture features

Defining texture is a difficult task. However, several methods have been proposed to describe the visual texture in a quantitative way. Considering that it is impossible to represent a texture as an array of all the pixels that compose it and their possible values, different feature extraction methodologies have been proposed. A number of authors have put their efforts to categorize such texture features extraction methods. A first proposal was introduced by Van Gool et al. [135], where the methods were grouped into statistical and structural methods. According to Tuceryan and Jain [130], the categorization may be extended into four major groups: geometrical, model-based, signal processing and statistical. This last categorization has been under polemics because there are methods that present properties from more than one category. Nevertheless, this categorization is the most accepted in the computer vision community.

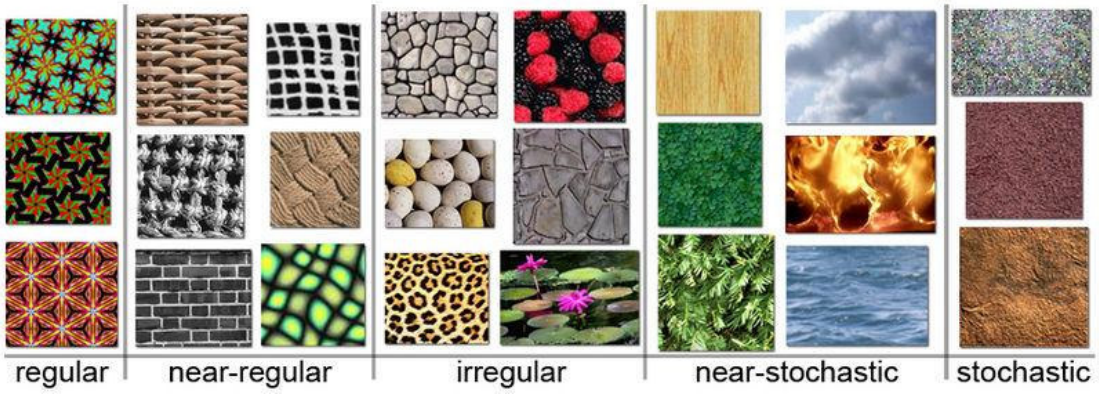


Figure 4.1: Types of textures arranged by texture regularity [69].

Geometrical methods

The methods that fall under the category of geometrical methods is characterized by their definition of texture as being composed of "texture elements" or texels. The method of analysis usually depends upon the geometric properties of such texture elements. The goal of these methods is to describe a texture with simple primitives, even if it is composed by complex structures. Geometrical methods are related to texture regularity, which ranges in a spectrum from periodic to stochastic. We view textures as a continuous spectrum where texture regularity varies gradually [69] (See Fig. 4.1).

On the one hand, regular textures are periodic patterns where the color and shape of all texels are repeating in equal intervals along two linearly independent directions. On the other hand, stochastic textures are considered as noisy. Different intensities are randomly scattered over the image, barely specified by the attributes of minimum and maximum brightness and average color. Textures observed on usually fall somewhere in-between these two extremes, emerging other categories like the group of near-regular textures. The near-regular group, comprises textures formed by texels that are not strictly identical, and can be observed on man-made and natural surfaces, e.g. textiles, floors, wallpapers, honeycombs. The irregularity observed in near-periodic textures can be caused by various statistical deviations along different dimensions [70], e.g. symme-

try, shape, noise, intensity and geometrical distortions.

Model-based methods

Model-based methods consist in creating a mathematical model able to describe the texture. The importance of such a model is not only based on its use for feature representation, but in its application in problems like texture synthesis. The model parameters capture the essential perceived qualities of texture.

One of the most important model-based methods is the Markov Random Fields. These models are able to capture the local information in an image assuming that the intensity of each pixel depends on the intensities of its neighbors. MRF models have been applied to various image processing applications such as texture synthesis [22], texture classification [9, 59], image segmentation [19, 126], image restoration [38] among others.

Signal processing methods

The signal processing methods are approaches based on some frequency analysis algorithm. These methods, are also called filtering approaches due to their application of a linear transform or a filter bank, followed by an energy measure.

Some of the most popular methods for texture analysis are the spatial domain filters [41, 103], the Fourier transform [8] and the use of Gabor filters [131, 18]. The use of spatial domain filters is the most direct way to capture image texture properties. Such methods define different kinds of textures by measuring the edge density per unit area. The Fourier domain filtering is the most popular method for frequency analysis of texture. As the psychophysical results indicated, the human visual system analyzes the textured images by decomposing the image into its frequency and orientation components [8].

Statistical methods

As it was aforementioned, texture may be regarded as the spatial distribution of gray values. Therefore, the use of statistical features is one of the pioneers and most-used methods in computer vision literature.

There is a considerable number of statistical methods for texture feature's extraction. Depending on the number of pixels defining the local feature, statistical methods can be further classified into first-order (one pixel), second-order (two pixels) and higher-order (three or more pixels) statistics. The basic difference is that first-order statistics estimate properties (e.g. average and variance) of individual pixel values, ignoring the spatial interaction between image pixels, whereas second- and higher-order statistics estimate properties of two or more pixel values occurring at specific locations relative to each other.

First-order texture measures are statistics calculated from the original image values and do not consider pixel neighborhood relationships. First-order statistic's analysis of texture is based on the intensity value concentrations on all or part of an image. Common features include moments such as mean, variance, dispersion, mean square value or average energy, entropy, skewness and kurtosis. The intensity levels of an image is thus a concise and simple summary of the statistical information contained in the image.

Among the second- and higher-order statistics, the most commonly used methodologies are:

- Gray Level Cooccurrence Matrix (GLCM). Originally proposed by Haralick [46], this matrix estimates texture properties related to second-order statistics. The matrix is conformed by records of differences of intensities of pairs of pixels, located at a given distance and orientation. Some features computed from the GLCM are the energy, entropy, contrast, homogeneity, among others.

- Autocorrelation function [47]. The autocorrelation function of an image can be

used to assess the amount of regularity, as well as the fineness/coarseness of the texture present in the image.

- Gray Level Sum and Difference Histograms (SDH). Proposed by Unser [134], these histograms were introduced as an alternative to the co-occurrence matrices used for texture analysis. The advantage of the SDH over the GLCM is the decreasing of the computation time and memory storage requirements.

- Non-parametric approaches. These approximations analyze textures at a micro level based on the calculation of local patterns. Examples of non-parametric approaches include the Local Binary Patterns (LBP) introduced by Ojala and Pietikainen [96] and the Coordinated Cluster Representation (CCR) proposed by Kurmyshev et al. [63].

4.2 Texture analysis problems

The various methods for modeling textures and extracting texture features can be applied in four broad categories of problems:

Texture classification

Texture classification is an open problem referring to the assignment of an unknown texture image into one of the several pre-learned texture classes. In general, texture image classification approaches are grouped as supervised or unsupervised, depending if there is a learning stage. In the unsupervised methods, the prior definition of classes is not required. These algorithms classify images in a predefined number of classes, using just the inherent information in the available data.

The supervised texture classification process involves two phases: the learning phase and the recognition phase. In the learning phase, the goal is to build a model for the texture content of each class present in the training data, which generally comprises of

images with known class labels. The texture content of the training images is captured with the chosen texture analysis method, which yields a set of textural features for each image. In the recognition phase, the texture content of an unknown sample is first characterized with the same texture analysis method. Then, the textural features of the sample are compared to those of the training images with a classification algorithm. Finally, the unknown sample is assigned to the category with the best match. Optionally, if the best match is not sufficiently good according to some predefined criteria, the unknown sample can be rejected.

Texture segmentation

As it was mentioned in Chapter 1, the segmentation of a given image may be accomplished using different kinds of attributes, texture in this case. Texture segmentation is one of the most challenging problems. The issue begins with the lack of a formal definition of textures, which explains the difficulty to conceptualize a model able to describe it. The human eye can easily recognize different textures, but it is quite difficult to define them in quantitative terms.

Texture synthesis

Texture synthesis is an active research area with wide applications in fields like computer graphics, image processing and computer vision. The texture synthesis problem can be stated as follows: given a finite sample texture, automatically create an outcome with similar visual attributes to the input and an arbitrarily defined size. Texture synthesis algorithms are intended to create an output image that meets the following requirements:

- The output should have the size given by the user.
- The output should be as similar as possible to the sample.

- The output should not have visible artifacts such as seams, blocks and misfitting edges.
- The output should not repeat, i.e., the same structures in the output image should not appear in multiple places.

Like most algorithms, texture synthesis should be efficient in computation time and in memory use.

Texture synthesis is a useful alternative way to create arbitrarily large textures. Furthermore, since it is only necessary to store a small sample of the desired texture, the synthesis can bring great benefits in memory storage. Most texture synthesis research has been focused on static textures. However, dynamic texture synthesis is receiving a growing attention during recent years.

Shape from texture

This problem is stated as the reconstruction of a 3D object from a 2D image. The 2D image is itself a repetition of texels and the apparent distortions of such texels are used to estimate the surface orientation of the 3D object. One of the first task to achieve the solution of this problem is to estimate the texel shape parameters.

4.3 Conducted research in texture analysis

Before the incorporation of texture features in an image segmentation framework, we have conducted intensive work within the the texture analysis research field. In order to remain focused on the segmentation problem, the main contributions of such work are going to be briefly presented in the following subsections. The studies developed were published in texture classification (Subsection 4.3.1), texture synthesis (Subsection 4.3.2) and shape from texture (Subsection 4.3.3) research areas.

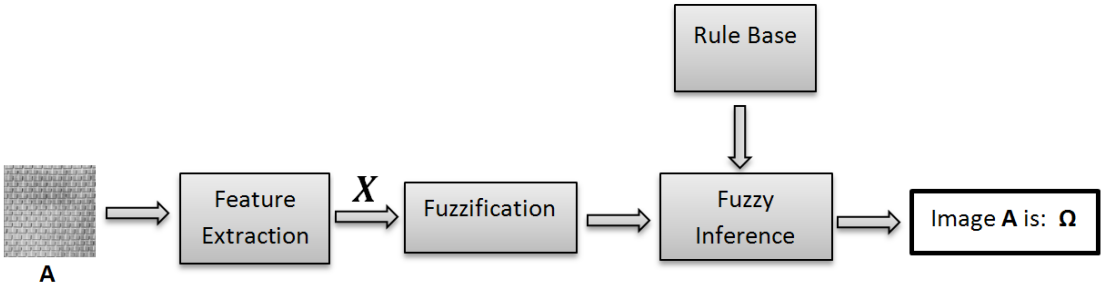


Figure 4.2: Operation of a fuzzy rule-based system for texture classification [73].

4.3.1 Visual texture classification using fuzzy inference

In this subsection, a supervised rule-based system for visual texture classification founded on fuzzy logic is briefly described. This approximation is, in general, built by two stages: learning and recognition. The learning phase mainly consists in the definition of the fuzzy if-then rules. This definition is based on a fuzzification of a set of quantitative features extracted from the given samples using a clustering approach. The set of rules obtained from the learning process is used in the recognition phase to obtain a class label. Rule-based systems allow us to represent the knowledge, and capture the expertise, in a set of if-then rules. In these rules, the premises are evaluated in order to obtain a conclusion. The recognition phase also consists in a feature extraction and a fuzzification step. Through a fuzzy inference process, the label assignation is achieved (See Figure 4.2).

In general, a fuzzy method for classification of textured images is proposed. The method is a multi-input and one output system, where the inputs are statistical textural features and the output is a label (class) assignation. A set of if-then type rules is built (one for each class) from known subimages of each class and the recognition is evaluated using unknown subimages from each class. A test texture sample is assigned to the rule whose activation level is the highest. Despite inherent difficulties in the classification problem, our system achieves good classification rates.

A fuzzy classifier is an alternative to the traditional systems, because of their tolerance to imprecision and uncertainty. In addition, we can highlight benefits by using fuzzy rules, since they can describe an image using linguistic terms. More technical details can be consulted in [73].

4.3.2 Dynamic texture synthesis with a spatiotemporal descriptor

Dynamic textures are essentially textures in motion, and have been defined as video sequences that show some kind of repetitiveness in time or space [40, 27]. Examples of these textures include recordings of smoke, foliage, and water in motion. Comparatively to the static texture synthesis, given a finite video sample of a dynamic texture, a synthesis method must create a new video sequence which looks perceptually similar to the input in appearance and motion.

In [71], we propose the use of local spatiotemporal patterns [142], as features in a non-parametric patch-based method for dynamic texture synthesis. The use of such features allows to capture the structure of local brightness variations in both spatial and temporal domains and therefore, describe the appearance and motion of dynamic textures. In our method, we take advantage of these patterns in the representation and selection of patches. With this improvement, we capture more structural information for a better patch matching, preserving properly the structure and dynamics of the given input sample. In this way, we can simplify the synthesis method with a very competitive performance in comparison with other patch-based methods.

The main contributions of this work are: (1) The extension of a patch-based approach, previously applied only for static texture, for its use in dynamic texture synthesis. (2) The dynamic texture description through local spatiotemporal features, instead of using only the color of the pixels. With this improvement we capture the local structural information for a better patch matching, preserving properly the appearance and dynamics of a given input texture. (3) A simplified method where the computation of an optimal seam between patches can be omitted. This can be achieved because of a

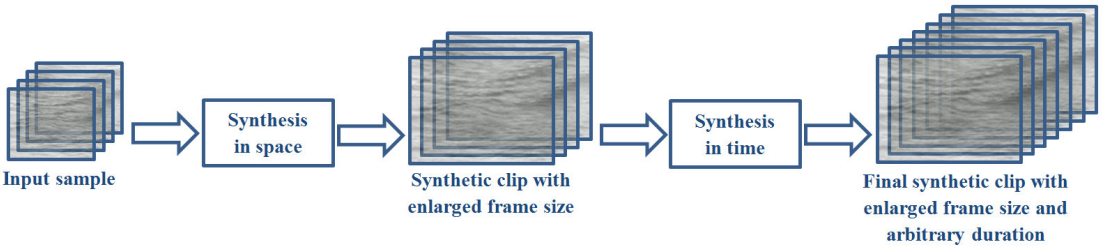


Figure 4.3: Final procedure to achieve the synthesis in both spatial and temporal domains [71].

better fitting and matching of patches. (4) A robust and flexible method that can cope with different kinds of dynamic textures videos, ranging from videos that show spatial and temporal regularity, those conformed by constrained objects and videos that contain both static and dynamic elements, showing irregularity in both appearance and motion. (5) A combination with a temporal domain synthesis method, in such a way that we can perform the synthesis in both the spatial and temporal domains. (6) The use of such method for video completion tasks.

The experiments show that the use of the local spatiotemporal representation outperforms other methods, without generating visible discontinuities or annoying artifacts. Results are evaluated using a double stimulus continuous quality scale methodology, which is reproducible and objective. We also introduce results for the use of our method in video completion tasks. Additionally, we hereby present that the proposed technique is easily extendable to achieve the synthesis in both spatial and temporal domains. More technical details can be consulted in the aforementioned paper [71].

4.3.3 Fast texel size estimation in visual texture using homogeneity cues

As it was mentioned in the introduction to this chapter, a structural approach establishes texture as an arrangement composed of ordered patterns, usually known as texture elements (texels) or textons, which are defined as the fundamental microstructure in

the image [143]. The texel size estimation has been very useful to address a number of problems, including shape from texture [77], texture recognition [66], texture synthesis [25, 26, 76, 70, 42], texture compression [85] and defect detection [94], among others. Furthermore, the texel can be used as a reference to improve the performance in classification tasks [54, 72, 67], segmentation [128, 82] and achieve scale invariant texture analysis [61, 141].

In the paper [74], a fast and robust approach to texel size detection in periodic and near-periodic textures is presented. The texel size estimation is addressed on periodic and near-periodic texture images. Such a problem has shown to be difficult when corrupted and distorted patterns are analyzed, and the accuracy and robustness are significant. In this study, the use of homogeneity cues computed using a difference histogram is proposed. The use of homogeneity cues for texture periodicity extraction has been specifically formulated for extracting horizontal and vertical displacement vectors, that correspond with a squared or rectangular tiling unit.

The main contributions of this work are: (1) The texel size can be automatically detected localizing the maximum value of the homogeneity cues. (2) The computation of such cues is not intricate nor time-consuming. (3) The method does not need a preprocessing step of the input. (4) Homogeneity cues are very effective and robust in identifying the texel size, even with noisy or deformed images. (5) This method can provide a very good approximation of texel size of near-regular textures. As an example taken from [74], we show an artificial and periodic texture with a tiling size unit of 72×42 pixels size in Fig. 4.4(a), and its corresponding homogeneity plots in Fig. 4.4(b) for both directions, horizontal and vertical. In Fig. 4.4(b) the periodic nature of both plots can be seen. The corresponding texel size is marked as a rectangle overlaid in Fig. 4.4(a), other three rectangles were added for comparison purposes. Notice that this tiling unit is repeated over the field of view, creating the texture image.

Experiments were carried out in order to evaluate the performance of this method. Results on artificially distorted images and on natural near-periodic images, show that

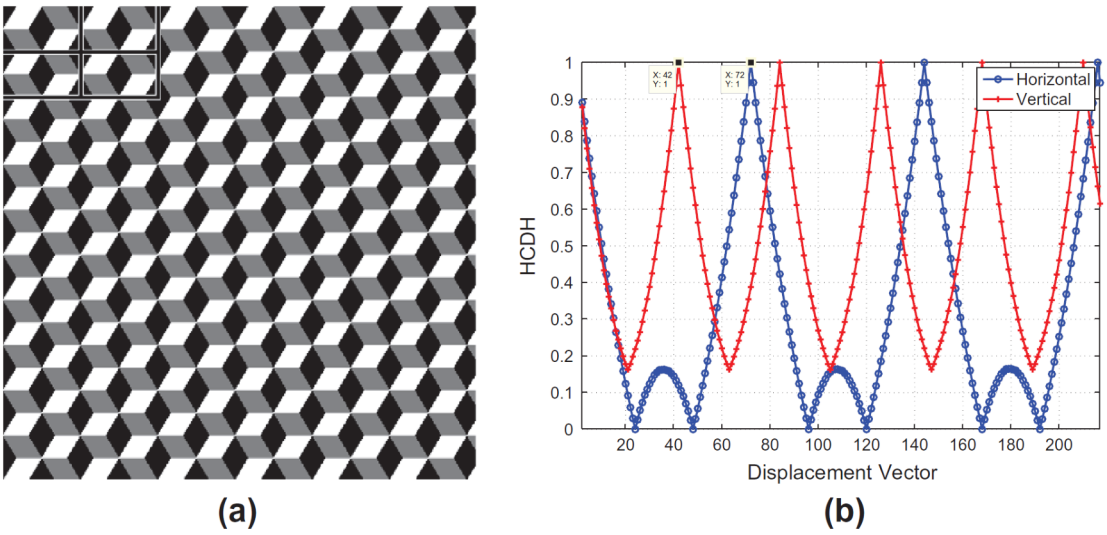


Figure 4.4: An artificial texture (a) with a texel of 72×42 pixels size and its homogeneity function (b) in both, horizontal and vertical directions [74].

the proposed approach is more accurate and robust than other state-of-the-art methods. Furthermore, the computation of homogeneity cues is not intricate nor time-consuming, and hence, it can be considered for practical applications where computation time is critical.

4.4 Concluding remarks of the chapter

The texture is one of the fundamental features used in the visual analysis of scenes. Texture is a property of all surfaces arising in many applications, such as in natural image segmentation. It is therefore, important to develop robust and efficient methods for processing textured images. In this chapter, we have reviewed the basic concepts and different methods for processing visual texture. Although an enormous research effort has been dedicated to the development of optimal texture analysis strategies, the problem of robust texture extraction is still an open issue. Depending of the final application, the appropriate integration of features in low-level tasks, as in image segmentation methods, is essential for the success of a given method.

The fundamental problems concerning the analysis of texture have been also introduced in this chapter, and the research conducted in the texture analysis field has been presented. A method using fuzzy inference for the classification of textures, the use of a spatiotemporal descriptor for dynamic texture synthesis and a fast texel size estimation in visual texture using homogeneity cues, have been briefly described.

The conducted research on this field, certainly helped for the main contribution of this dissertation, which consists on the incorporation of texture and color cues in a methodology based on rough set theory for segmentation tasks. Details of this method are presented in the next chapter.

CHAPTER 5

Integration of color and texture cues in a rough set-based segmentation method

In this chapter, a proposal for the integration of features in a rough set-based segmentation approach using color and texture cues (from now on, this method is referred to as RCT), is presented. The rough set-based methodology takes advantage of the fact that it includes spatial information about the pixels and associates them with their neighbors with similar features. In this proposal, the color is represented in a perceptual color space, while the texture features are computed using a standard deviation map that records intensity variations in a given neighborhood.

This chapter is organized as follows. In Section 5.1 the proposed segmentation framework is presented, starting with an overall description of the method. Additionally, in this section, our feature extraction scheme and its implementation in the proposed approach are also introduced. The experiments and results are given in Section 5.2, followed by the concluding remarks in Section 5.3.

5.1 Proposed segmentation framework

The process of the proposed segmentation approach is illustrated in Fig. 5.1. In general, it is separated into three main steps: feature extraction, the pre-segmented image obtention and finally, the application of a region merging step, in order to diminish over-segmentation issues. First, a color space transformation from the RGB space to the $CIELab$ perceptual space is applied to the input image, using the same equations presented in Section 3.1.1 from Chapter 3. The texture features T, described in the following section, are extracted from the lightness component L in the $CIELab$ space. Hence, the color components a and b and the texture features T are represented in separate channels. After that, the rough set-based segmentation approach is performed, and the pre-segmented image is obtained. This approach allows the association of the feature information in a local neighborhood, and makes the segmentation fully unsupervised. Specifically, in this step, it is important to mention that Mushrif and Ray [91] have proposed to use $C = \{R, G, B\}$ as the three channels of information. In our color-alone approximation described in Chapter 3, we have changed the image representation to the three channels $C = \{L, a, b\}$ and in the proposal presented in this chapter, the image representation is carried out as $C = \{T, a, b\}$. At the end of our method, a region merging step is performed on the union of the three outcomes, reducing over-segmentation. The color space transformation, the rough set-based segmentation and the region merging blocks from Fig. 5.1 are the same as the ones described in Sections 3.1.1, 3.1.2 and 3.1.3 from Chapter 3, respectively. In the following section, the use of a standard deviation map as a texture feature is presented.

5.2 A standard deviation map as texture feature

The pixels in textured regions of an image show more intensity variations than the pixels in homogeneous regions therefore, a measure of those variations, e.g. the standard

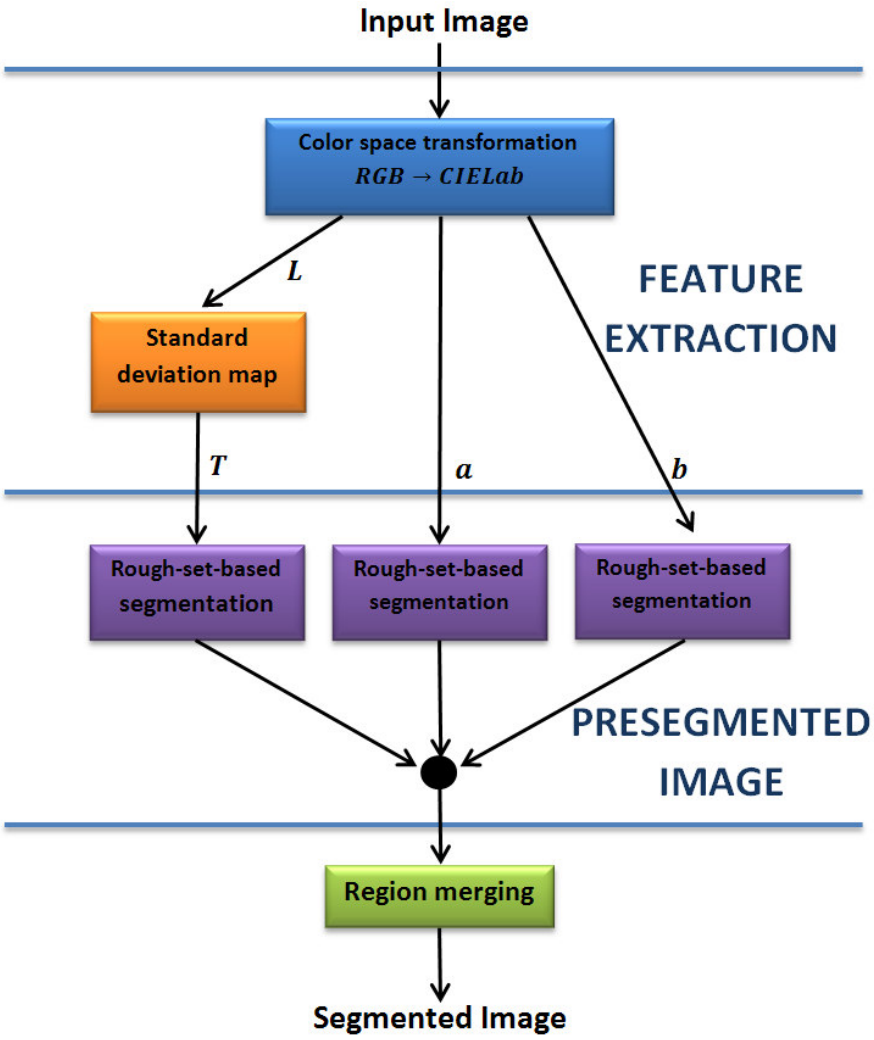


Figure 5.1: The general process of the proposed segmentation approach, RCT.

deviation, may be used to determine such textured regions. In this regard, a standard deviation map is obtained as follows: for each pixel in the image, the standard deviation σ of the pixel intensities in a neighborhood is calculated. This neighborhood consists of a square region containing $k = (2d + 1)^2$ pixels, and centered at the current pixel position. The parameter d is the number of pixels from the central pixel to a side of the window. Equations 5.1 and 5.2 are used to obtain σ

$$\mu_r = \frac{1}{k} \sum_{i=1}^{k} x_i^r \tag{5.1}$$

$$\sigma = \sqrt{\mu_2 - \mu_1^2} \tag{5.2}$$

where x is the intensity value for the i-th pixel of the neighborhood, and μ_1 and μ_2 are the first and second statistical moments around zero, respectively. Special considerations are required for those pixels close to the image edges, where the corresponding neighborhood extends beyond the image boundaries. If the pixels outside the image are considered to be of intensity zero, false texture variations appear at the edges of the standard deviation map. To reduce these artifacts, the intensities of the pixels outside the image are matched to the intensity of the pixels inside the image that are located at the same distance from the edge. This mirroring procedure preserves the texture characteristics of the regions near to the image edges.

Examples of the resulting standard deviation maps using a $d = 10$ are shown in Fig. 5.2. It can be noticed from these examples that the different texture regions are shown in gray levels. The near-black regions correspond to regions of homogeneous intensity in the original image. The gray levels are in function of the intensity variations of the texture. These examples show that this feature is powerful enough to distinguish the textured regions within an image.

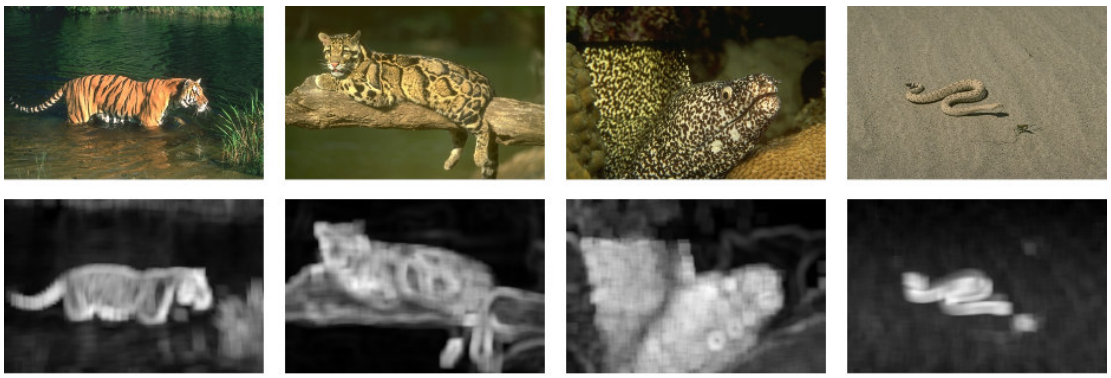


Figure 5.2: Examples of the resulting standard deviation maps with a $d = 10$.

5.3 Experiments and Results

In this section, the experiments conducted on natural scene images in order to evaluate the performance of our proposal, are presented. The evaluation is accomplished by a thorough qualitative and quantitative analysis. Additionally, the assessment of the RCT in comparison to other similar and recent state-of-the-art approaches is also presented. The corresponding parameters were empirically tuned, following the same methodology described in the section 3.3.1 in Chapter 3.

In order to revisit our method and illustrate it, an example of the resulting images in each step and the final segmented image are presented in Fig. 5.3. In Figs. 5.3a, 5.3b and 5.3c, we can see the images of the feature channels used in RCT: a, b and T, respectively. The pre-segmented image is shown in Fig. 5.3d. Such an image is the result of the rough set-based segmentation, before to its submission to the merging process. The Fig. 5.3e shows the resulting segmentation of RCT after the merging process, where the over-segmentation issues were significantly reduced. In this image, we can see that the pixels of the flower petals are associated in one segment despite the intensity differences of the red color. Moreover, the RCT method is able to clearly separate the red petals of the flower from the yellow center and the green background.

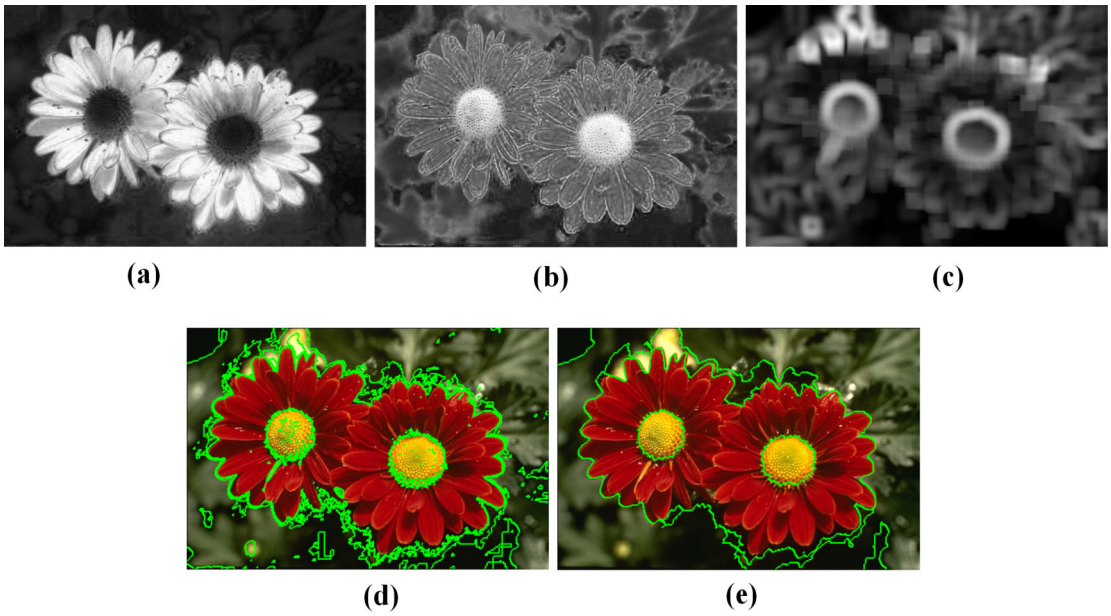


Figure 5.3: Resulting images of the inner process. Feature components (a) a, (b) b and (c) T, respectively. (d) The pre-segmented image before the region merging process and (e) the final segmented image.

Table 5.1: Performance analysis of the RCT method using the three measures.

	GCE	BDE	PRI
Mean (μ)	0.175	8.007	0.785
Std. Dev. (σ)	0.105	4.450	0.121

For the evaluation of the RCT method, the scenario described in Chapter 1 for the assessment of segmentation algorithms is replicated, where the Berkeley Segmentation Data Set (BSD) [83] is used and the quantitative evaluation is carried using the three metrics: Probabilistic Rand Index (PRI) [132], Global Consistency Error (GCE) [83] and the Boundary Displacement Error (BDE) [34].

The distributions of the three measures GCE, BDE and PRI for the 300 images in the BSD are presented in Fig. 5.4. This figure shows the frequency at which the RCT method attains a given value in each quantitative measure. For the PRI distribution (Fig. 5.4a), one can note the leaning of the PRI distribution to the maximum value of 1. This means that a high number of images achieve high PRI values, implying that the outcomes using the RCT method have a correspondence with the ground-truth. This figure also shows the tendency of the GCE (Fig. 5.4b) and BDE (Fig. 5.4c) distributions, where the tendency towards an error of zero is noticeable. The corresponding mean performance and the standard deviation for each quantitative measure GCE, BDE and PRI are shown in Table 5.1.

A comparison of the performance, in terms of the quantitative measures against other recent and similar approaches, was conducted. The first comparison is performed against the color-based technique proposed by Mushrif [92] using A-IFS histon that uses the *RGB* color representation. In the original A-IFS histon based article, the average performance is assessed using only the PRI measure. The average performance reported for the A-IFS method is 0.7706, which is lower than the 0.785 achieved by our RCT. This implies that the image representation in perceptual color spaces and the inclusion of texture information improves the performance of rough set-based methodologies.

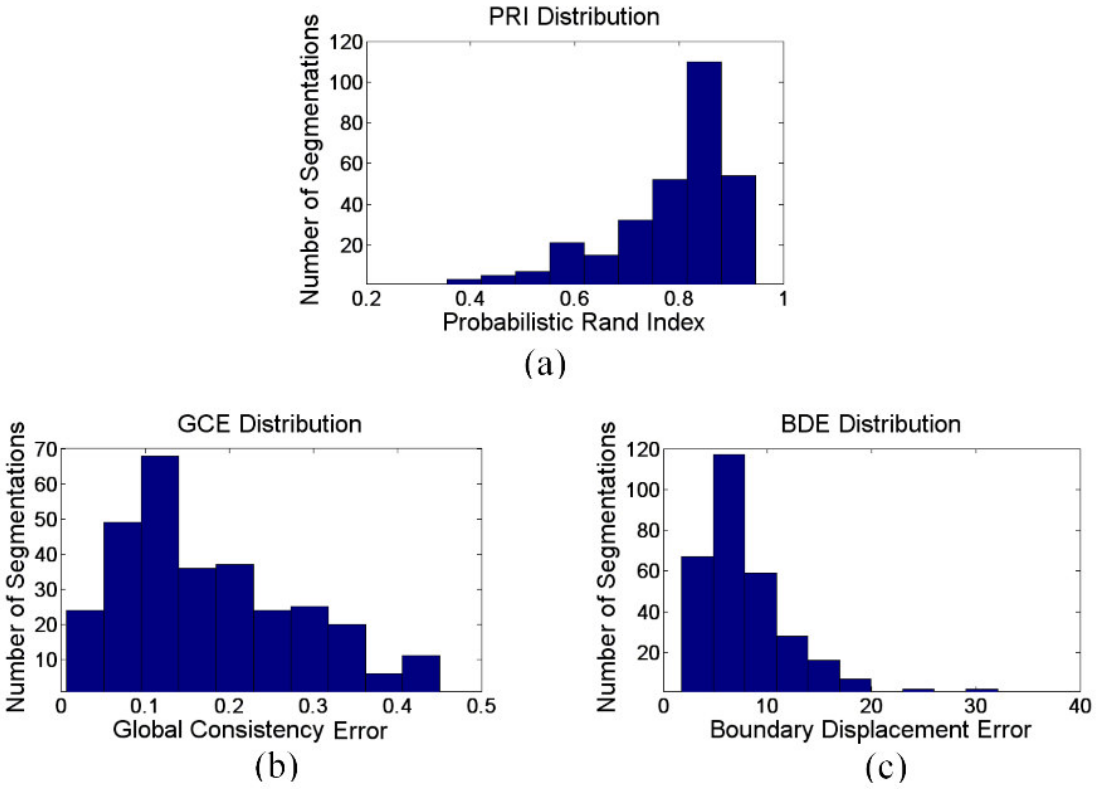


Figure 5.4: Distribution of the resulting values of the three measures for the Berkeley Data set. For (a) a higher value is better and for (b) and (c) a smaller value is better.

Table 5.2: Average performance and comparison with other methods. Within the parentheses is the number of images used by the authors for their evaluation.

	GCE	BDE	PRI
JSEG (300)	0.196	8.960	0.774
CTM (300)	0.18	9.490	0.760
CSC (100)	0.225	8.634	**0.796**
RCT (300)	**0.175**	**8.007**	0.785

A comparison with other three state-of-the-art algorithms that use both color and texture features was also carried out in our evaluation: the JSEG method proposed by Deng et al. [24], the CTM approach introduced by Yang et al. [138] and the method presented by An and Pun [2] referred to as CSC.

The average performance of each method using the three quantitative measures is presented in Table 5.2. For the JSEG and the CTM approaches, the source code provided at their web pages were executed with the 300 BSD images. For the CSC method, the results presented by the authors were taken from the original article. This table shows that the best results for the GCE and the BDE measures are achieved by the proposed RCT method. This means that our method is more accurate, since it has the lowest errors in relation to the ground-truth segmentations. Comparing methods using the PRI measure, the method RCT has obtained a 0.785 score, outperforming the JSEG (0.774) and CTM (0.760) approaches. The method that attains the highest PRI value is the CSC, reporting an average PRI of 0.796.

It is important to highlight that the CSC evaluation, as mentioned in the original article [2], has been carried out using only a subset of 100 images from the complete set of 300 images from the BSD. Since the authors of such CSC method do not define the selected subset of images used in their experiments, the tendency of the mean achieved by our RCT is analyzed for different subset sizes. For this analysis, we have taken the $K = \{1, 2, \ldots, 300\}$ images from the RCT results that achieve the best performance for the PRI measure. The PRI mean tendency is presented in Fig. 5.5, where it is shown

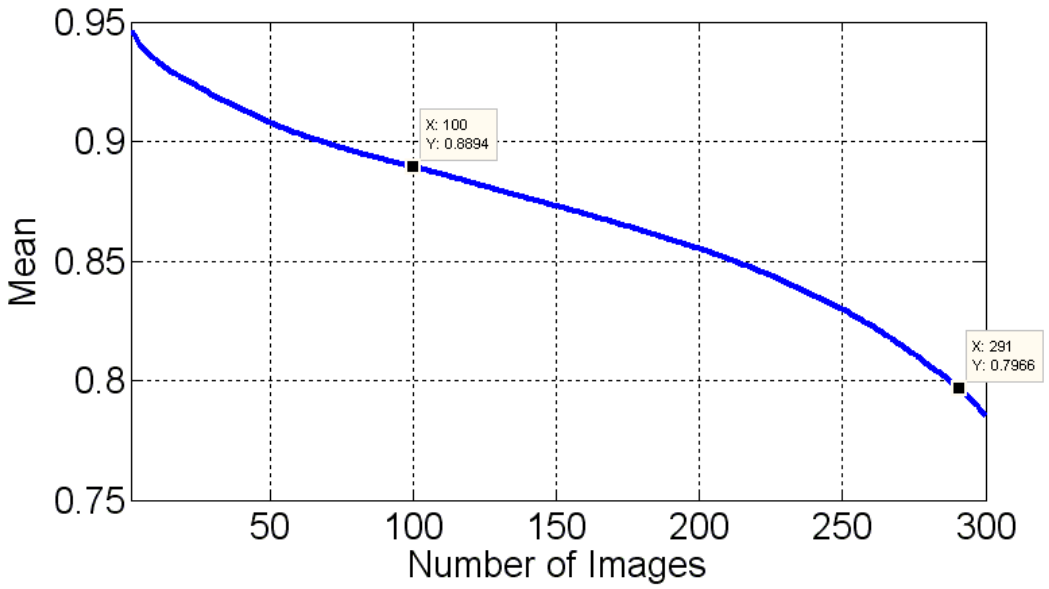


Figure 5.5: Progress of the PRI measure mean achieved by the RCT method depending of the number of images taken from the BSD.

that if the number of testing images is increased, the average performance decreases. In this figure we can see that if 291 images are used, the RCT method achieves the mean performance of the CSC method of 0.796. Furthermore, if the best 100 images are taken for the RCT evaluation, our method achieves a mean PRI value of 0.8894, a much higher value than the 0.796 obtained by the CSC method.

A qualitative comparison of the segmentation results is presented in Fig. 5.6, where the first column corresponds to the CTM outcomes, the second column shows the resultant images of the JSEG method, and our RCT segmentations are shown in the third column. We only present qualitative comparisons of the methods whose results are available. In these examples, the edges of the segments are overlaid on the original image. From this qualitative comparison, we can observe that the RCT method is in general able to associate pixels with similar color and texture in single segments. For example in Fig. 5.6a, there are over-segmentation issues in the outcomes of the CTM

and JSEG methods. In Fig. 5.6b a challenging image is shown, since the color of the cheetah and its background are very similar. For this case, only the CTM and the RCT methods are able to separate the differences in texture of this image. In Fig. 5.6c one can notice the ability of RCT for the association of the different intensities of blue in the sky, while the other methods separate them in two (Fig. 5.6c2) or even in three different segments (Fig. 5.6c1). This ability can be also appreciated in Fig. 5.6d, where the RCT is able to merge the elephants in only one segment. In the case of Fig. 5.6e and Fig. 5.6f the performance of the three methods is very similar. The last example shows how the CTM and the RCT succeed in the association of the pixels of the kangaroo while the JSEG attains an over-segmentation.

5.4 Concluding remarks of the chapter

In this chapter, the integration of color and visual texture cues in a rough set theory-based segmentation approach has been proposed. Some advantages have been identified in comparison to other methods. The proposed RCT approach considers the spatial correlation and similarity of neighboring pixels, including information of both color and texture. Moreover, an important advantage over the clustering methods is that the RCT methodology does not require cluster initialization because the number of segments is automatically estimated for each image. The RCT method computes the similarity between pixels within a neighborhood, using both their color and textural information. These features have shown to be simple to compute and yet representative of the image information. A thorough analysis of the performance of the RCT method showed that it can be successfully applied to natural image segmentation, where the resemblance to the human perception may be desirable.

In general, it is observed that the use of a perceptual color representation, and the addition of textural features attains best results than using only color cues. Additionally, the presented modifications, like the significant peaks' selection jointly to the merging

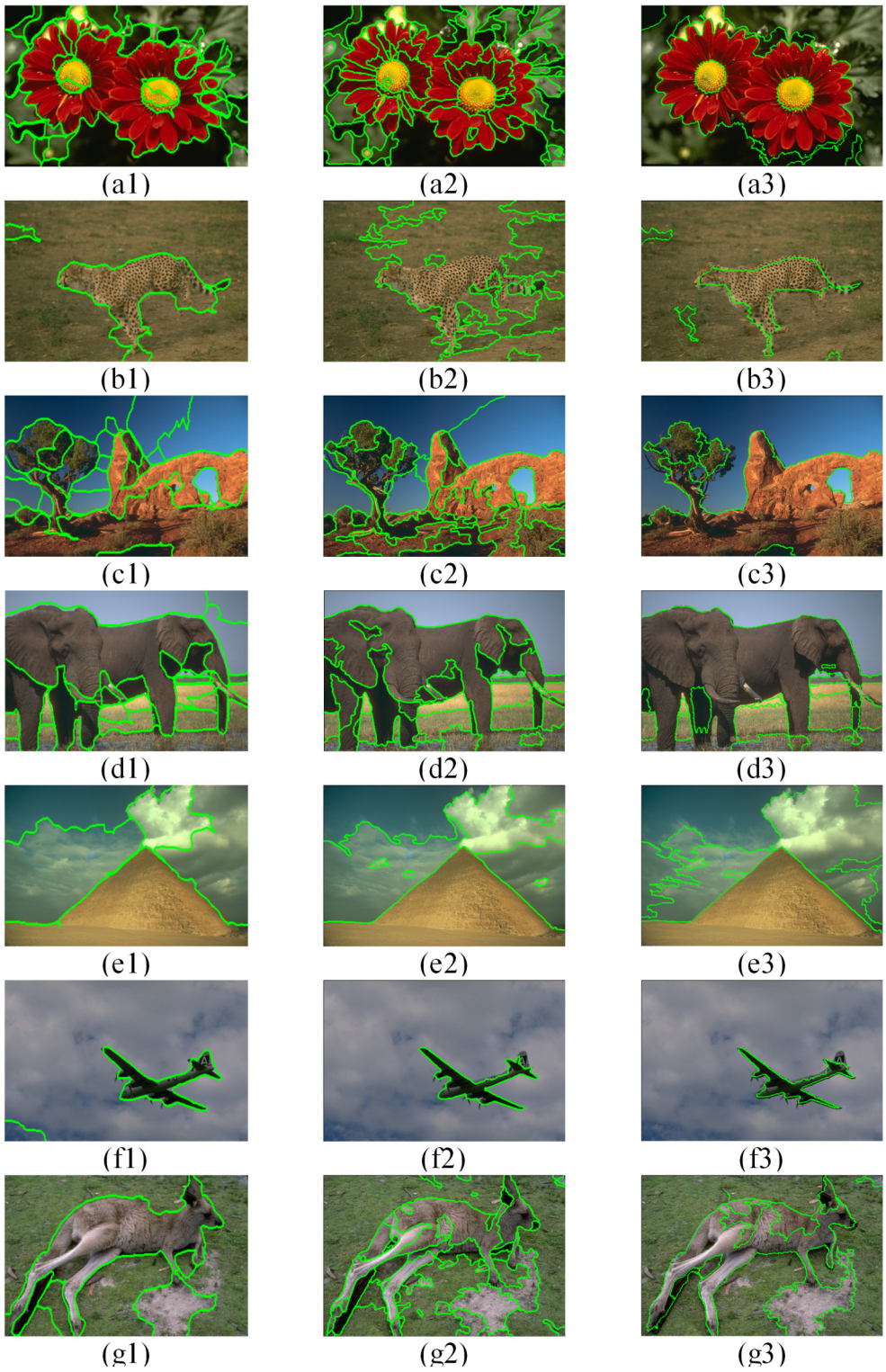


Figure 5.6: A qualitative comparison of 7 out of 300 segmentation results between the CTM (first column), JSEG (second column) and our RCT (third column). The borders of the segments are overlaid on the original image.

strategy, allow best results in comparison to other state-of-the-art methods in terms of a set of quantitative measures.

General Conclusions and Perspectives

Segmentation of natural images is a challenging task because the color and texture attributes are not uniformly distributed and as a result, the process of identifying the homogeneous regions in the image is very difficult. In this regard, the use of computational intelligence tools is proposed in this thesis. Such techniques, in contrast to classical hard computing techniques, are tolerant of imprecision, uncertainty and partial truth, typical characteristics of natural images. Specifically, among the computational intelligence tools, we apply the rough sets' concepts.

In this study, a set of modifications were proposed in order to improve a rough set-based segmentation method for color-alone segmentation. In addition, we explore the use of both color and texture features in such improved methodology. Main contributions of the proposed color-alone segmentation approach are twofold. First, by using an adaptive threshold selection, the approach is automatically adjustable according to the image content. Second, a region merging process, which takes into account both features and spatial relations of the resulting segments, let us minimize over-segmentation issues. These two improvements lead our method to overcome some performance issues shown by the previous rough set theory-based approaches. In addition, a study to determine the best suited color representation for this approach was carried out, concluding that the best results were obtained using perceptually uniform color spaces. The use of a perceptual color representation allows our system to improve the performance

in the association of similar colors. The evaluation was accomplished by a thorough qualitative and quantitative analysis. Experiments on an extensive database show that these modifications result in a method with better outcomes, outperforming other rough set-based approaches and classic reference segmentation algorithms used for color-alone segmentation.

After the color-alone segmentation was achieved, the integration of color and visual texture cues in the rough set theory-based segmentation approach has been proposed. This approach computes the similarity between pixels within a neighborhood, using both their color and textural information. The color cues correspond to the a and b channels of the $CIELab$ color space, and the texture feature is computed as a standard deviation map. These features have shown to be simple to compute and yet representative of the image information. A thorough analysis of the performance of the proposed method showed that it can be successfully applied to natural image segmentation, where the resemblance to human perception may be desirable. Furthermore, a quantitative evaluation shows that the synergistic integration of features in this framework results in better segmentation outcomes, in comparison to those obtained by other related and state-of-the-art methods proposed for the same task.

Perspectives and future work

Different perspectives, ideas and future work arise from the development of this methodology. The first opportunity area in this framework is the parameter optimization. In this thesis, the best parameters are found with an empirical methodology with an exhaustive search. Such optimization may be improved using an evolutionary computation algorithm. In the computation of thresholds and similarity distances, it is possible to include a fuzzy logic approach, which gives a mechanism to represent the ambiguity within an image.

Other directions of future research can be focused on improving the computational time associated with the proposed method. The most computationally intensive com-

ponent is the region merging, whose implementation can be optimized.

Further work may involve the use of the proposed framework for different high-level tasks, for example: its application to the segmentation of medical data, visual navigation of a robot, or its use in the development of content based image retrieval systems.

Publications resulting from this research

The publications resulting from this research are enumerated below:

1. R. A. Lizarraga-Morales, R.E. Sanchez-Yanez, V. Ayala-Ramirez. Periodicity and texel size estimation of visual texture using entropy cues, Computación y Sistemas. 14(3): 309-319 (2011).

2. R. A. Lizarraga-Morales, R.E. Sanchez-Yanez, V. Ayala-Ramirez. Homogeneity cues for texel size estimation of visual texture. J. F. Martinez-Trinidad et al. (Eds.): MCPR 2011, LNCS 6718, pp. 220-229, 2011.

3. R. A. Lizarraga-Morales, R.E. Sanchez-Yanez, V. Ayala-Ramirez, Visual texture classification using fuzzy inference, Proc. Tenth Mexican International Conference on Artificial Intelligence (MICAI 2011), pp. 150-154, Puebla, Mexico, Nov 26-Dec 04, 2011.

4. R. A. Lizarraga-Morales, Y. Guo, G. Zhao and M. Pietikäinen. Dynamic Texture Synthesis in Space with a Spatio-Temporal Descriptor. J.-I. Park and J. Kim (Eds.): ACCV 2012 Workshops, Part I, LNCS 7728, pp. 38-49, 2013.

5. R. A. Lizarraga-Morales, R. E. Sanchez-Yanez and V. Ayala-Ramirez. Fast texel size estimation in visual texture using homogeneity cues. Pattern Recognition Letters. 34(1):414-422. (2013).

6. R. A. Lizarraga-Morales, R. E. Sanchez-Yanez, V. Ayala-Ramirez and A. J. Patlan-Rosales. Improving a rough set theory-based segmentation approach using adapt-

able threshold selection and perceptual color spaces. Accepted to be published in Journal of Electronic Imaging (SPIE).

7. R. A. Lizarraga-Morales, R. E. Sanchez-Yanez, V. Ayala-Ramirez and F. E. Correa-Tome. Integration of Color and Texture Cues in a Rough-Set-based Segmentation Method. Accepted to be published in Journal of Electronic Imaging (SPIE).

Bibliography

[1] M. Abdulghafour. Image segmentation using fuzzy logic and genetic algorithms. *Journal of WSCG*, 11(1), 2003.

[2] N.-Y. An and C.-M. Pun. Color image segmentation using adaptive color quantization and multiresolution texture characterization. *Signal, Image and Video Processing*, pages 1–12, May 2012. doi=10.1007/s11760-012-0340-2.

[3] J. Angulo. Morphological texture gradients. application to colour+texture watershed segmentation. In *International Symposium on Mathematical Morphology*, pages 399–410. 2007.

[4] J.-P. Aribot. Texture segmentation. Technical report, Swiss Federal Institute of Technology, 2003.

[5] I. Bloch. On links between mathematical morphology and rough sets. *Pattern Recognition*, 33(9):1487–1496, 2000.

[6] L. Busin, J. Shi, N. Vandenbroucke, and L. Macaire. Color space selection for color image segmentation by spectral clustering. In *Proc. of the IEEE Int. Conf. on Signal and Image Processing Applications*, pages 262–267, 2009.

[7] L. Busin, N. Vandenbroucke, and L. Macaire. Color spaces and image segmentation. *Advances in Imaging and Electron Physics*, 151(7):65–168, 2008.

[8] F. W. Campbell and J. Robson. Application of fourier analysis to the visibility of gratings. *The Journal of Physiology*, 197(3):551, 1968.

[9] R. Chellappa and S. Chatterjee. Classification of textures using gaussian markov random fields. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 33(4):959–963, 1985.

[10] J. Chen, G. Zhao, M. Salo, E. Rahtu, and M. Pietikainen. Automatic dynamic texture segmentation using local descriptors and optical flow. *IEEE Transactions on Image Processing*, 22(1):326–339, 2013.

[11] K.-M. Chen and S.-Y. Chen. Color texture segmentation using feature distributions. *Pattern Recognition Letters*, 23:755–771, 2002.

[12] P. Chen and T. Pavlidis. Segmentation by texture using a co-occurence matrix and a split-and-merge algorithm. *Computer Graphics and Image Processing*, 10:172–182, 1979.

[13] Y. S. Chen, B. T. Chen, and W. H. Hsu. Efficient fuzzy c-means clustering for image data. *Journal of Electronic Imaging*, 14(1):013017–013017–13, 2005.

[14] H. Cheng, X. Jiang, Y. Sun, and J. Wang. Color image segmentation: advances and prospects. *Pattern Recognition*, 34(12):2259–2281, 2001.

[15] H. Cheng, X. Jiang, and J. Wang. Color image segmentation based on homogram thresholding and region merging. *Pattern Recognition*, 35(2):373 – 393, 2002.

[16] P. Chiranjeevi and S. Sengupta. Robust detection of moving objects in video sequences through rough set theory framework. *Image and Vision Computing*, 30:829–842, 2012.

[17] K.-S. Chuang, H.-L. Tzeng, S. Chen, J. Wu, and T.-J. Chen. Fuzzy c-means clustering with spatial information for image segmentation. *Computerized Medical Imaging and Graphics*, 30(1):9 – 15, 2006.

[18] M. Clark, A. C. Bovik, and W. S. Geisler. Texture segmentation using gabor modulation/demodulation. *Pattern Recognition Letters*, 6(4):261–267, 1987.

[19] F. S. Cohen and D. B. Cooper. Simple parallel hierarchical and relaxation algorithms for segmenting noncausal markovian random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (2):195–219, 1987.

[20] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002.

[21] F. Correa-Tome, R. Sanchez-Yanez, and V. Ayala-Ramirez. Comparison of perceptual color spaces for natural image segmentation tasks. *Optical Engineering*, 50(11):117203, 2011.

[22] G. R. Cross and A. K. Jain. Markov random field texture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1):25–39, 1983.

[23] L. S. Davis and A. Mitiche. MITES: A model-driven, iterative texture segmentation algorithm. *Computer Graphics and Image Processing*, 19(2):95 – 110, 1982.

[24] Y. Deng and B. Manjunath. Unsupervised segmentation of color-texture regions in images and video. *IEEE Transactions on Pattern Analysis and Machine Vision Intelligence*, 23(8):800–810, August 2001.

[25] K. Djado, R. Egli, and F. Deschênes. Extraction of a representative tile from a near-periodic texture. In *Proc. of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, GRAPHITE '05, pages 331–337, 2005.

[26] W. Dong, N. Zhou, and J.-C. Paul. Tile-based interactive texture design. In *Proc. of the 3rd International Conference on Technologies for E-Learning and Digital Entertainment*, Edutainment '08, pages 675–686, 2008.

[27] G. Doretto, E. Jones, and S. Soatto. Spatially homogeneous dynamic textures. In *Computer Vision - ECCV 2004*, volume 3022 of *LNCS*, pages 591–602. 2004.

[28] M. Dorigo and G. Di Caro. Ant colony optimization: a new meta-heuristic. In *Proc. of the Congress on Evolutionary Computation*, volume 2. IEEE, 1999.

[29] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, New York, 1973.

[30] R. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proc. of the Sixth International Symposium on Micro Machine and Human Science*, pages 39–43. IEEE, 1995.

[31] M. Fairchild. *Color Appearance Models*. Wiley, Hoboken, NJ, 2nd edition, 2005.

[32] J. Fan. Automatic image segmentation by integrating color-edge extraction and seeded region growing. *IEEE Transactions on Image Processing*, 10(10):1454–1466, Oct 2001.

[33] I. Fondón, C. Serrano, and B. Acha. Color-texture image segmentation based on multistep region growing. *Optical Engineering*, 45(5):057002–057002, 2006.

[34] J. Freixenet, X. Munoz, D. Raba, J. Marti, and X. Cufi. Yet another survey on image segmentation: Region and boundary information integration. In A. H. et al., editor, *Proc. of the European Conference on Computer Vision, LNCS 2352*, pages 408–422. Springer-Verlag Berlin Heidelberg, 2002.

[35] K.-S. Fu and J. Mui. A survey on image segmentation. *Pattern Recognition*, 13(1):3–16, 1981.

[36] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, Jan 1975.

[37] L. Garcia Ugarriza, E. Saber, S. Vantaram, V. Amuso, M. Shaw, and R. Bhaskar. Automatic image segmentation by dynamic region growth and multiresolution merging. *IEEE Transactions on Image Processing*, 18(10):2275–2288, 2009.

[38] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):721–741, 1984.

[39] N. I. Ghali, W. G. Abd-Elmonim, and A. E. Hassanien. Object-based image retrieval system using rough set approach. In *Advances in reasoning-based image processing intelligent systems*, volume 29, pages 315–329, 2012.

[40] B. Ghanem and N. Ahuja. Phase PCA for dynamic texture video compression. In *Proc. of the IEEE International Conference on Image Processing*, volume 3, pages 425–428, 2007.

[41] J. J. Gibson. *The perception of the visual world.* Houghton Mifflin, 1950.

[42] Y. Gui and L. Ma. Periodic pattern of texture analysis and synthesis based on texels distribution. *The Visual Computer*, 26(6-8):951–964, June 2010.

[43] P. Guo and M. Lyu. A study on color space selection for determining image segmentation region number. In *Proc. of International Conference on Artificial Intelligence*, volume 3, pages 1127–1132, 2000.

[44] S. Han, W. Tao, D. Wang, X. Tai, and X. Wu. Image segmentation based on grabcut framework integrating multiscale nonlinear structure tensor. *IEEE Transactions on Image Processing*, 18(10):2289–2302, 2009.

[45] R. Haralick. *Fundamentals in Computer Vision: An Advanced Course*, chapter Image Segmentation Survey, pages 209–223. Cambridge University Press, 1983.

[46] R. Haralick, K. Shanmugan, and I. Dinstein. Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6):610–621, 1973.

[47] R. M. Haralick. Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67(5):786–804, 1979.

[48] R. M. Haralick and L. G. Shapiro. Image segmentation techniques. *Computer Vision, Graphics, and Image Processing*, 29(1):100 – 132, 1985.

[49] A. Hassanien. Fuzzy rough sets hybrid scheme for breast cancer detection. *Image and Vision Computing*, 25(2):172–183, 2007.

[50] A. E. Hassanien, A. Abraham, J. F. Peters, G. Schaefer, and C. Henry. Rough sets and near sets in medical imaging: A review. *IEEE Transactions on Information Technology in Biomedicine*, 13(6):955–968, 2009.

[51] J. H. Holland. *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press, 1975.

[52] D. E. Ilea and P. F. Whelan. CTex- An adaptive unsupervised segmentation algorithm based on color-texture coherence. *IEEE Transactions on Image Processing*, 17(10):1926–1939, 2008.

[53] D. E. Ilea and P. F. Whelan. Image segmentation based on the integration of colour-texture descriptors- A review. *Pattern Recognition*, 44:2479–2501, 2011.

[54] S.-R. Jan and Y.-C. Hsueh. Window-size determination for granulometrical structural texture classification. *Pattern Recognition Letters*, 19(5-6):439–446, Apr. 1998.

[55] S. Ji and H.-W. Park. Image segmentation of color image based on region coherency. In *Proc. of the IEE International Conference on Image Processing*, volume 1, pages 80–83. IEEE, 1998.

[56] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):881–892, Jul 2002.

[57] A. Kapoor, P. Pandey, and K. K. Biswas. Fuzzy rule based document image segmentation for component labeling. In *Proc. of the Third National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*, pages 11–14, 2011.

[58] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[59] A. Khotanzad and R. L. Kashyap. Feature selection for texture recognition based on image synthesis. *IEEE Transactions on Systems, Man and Cybernetics*, 17(6):1087–1095, 1987.

[60] J. Kim and K. Hong. Colour-texture segmentation using unsupervised graph cuts. *Pattern Recognition*, 42(5):735–750, 2009.

[61] G. Kishor, D. Mital, and W. Goh. Invariant texture analysis based on attributes of texture elements. In *Proc. of the IEEE International Symposium on Industrial Electronics*, volume 1 of *ISIE '95*, pages 400–404, 1995.

[62] J. Komorowski, Z. Pawlak, L. Polkowski, and A. Skowron. Rough sets: A tutorial, 1998.

[63] E. Kurmyshev and M. Cervantes. A quasi-statistical approach to digital image representation. *Revista Mexicana de Fisica*, 42(1):104–116, 1996.

[64] F. Kurugollu, B. Sankur, and A. Harmanci. Color image segmentation using histogram miltithresholding and fusion. *Image and Vision Computing*, 19(13):915–928, 2001.

[65] K. I. Laws. Textured image segmentation. Technical report, DTIC Document, 1980.

[66] T. Leung and J. Malik. Recognizing surfaces using three-dimensional textons. In *Proc. of the 7th IEEE International Conference on Computer Vision*, volume 2, pages 1010–1017, 1999.

[67] Z. Li, G. Liu, Y. Yang, and J. You. Scale- and rotation-invariant local binary pattern using scale-adaptive texton and subuniform-based circular shift. *IEEE Transactions on Image Processing*, 21(4):2130–2140, April 2012.

[68] Y. W. Lim and S. U. Lee. On the color image segmentation algorithm based on the thresholding and the fuzzy c-means techniques. *Pattern Recognition*, 23(9):935 – 952, 1990.

[69] W.-C. Lin, J. H. Hays, C. Wu, V. Kwatra, and Y. Liu. Quantitative evaluation on near regular texture synthesis. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 427–434, June 2006.

[70] Y. Liu, W.-C. Lin, and J. Hays. Near-regular texture analysis and manipulation. *ACM Transactions on Graphics*, 23(3):368–376, Aug. 2004.

[71] R. Lizarraga-Morales, Y. Guo, G. Zhao, and M. Pietikäinen. Dynamic texture synthesis in space with a spatio-temporal descriptor. In J.-I. Park and J. Kim, editors, *Asian Conference on Computer Vision 2012 Workshops, Part I*, volume 7728 of *LNCS*, pages 38–49, 2013.

[72] R. Lizarraga-Morales, R. Sanchez-Yanez, and V. Ayala-Ramirez. Optimal spatial predicate determination of a local binary pattern. In *Proc. of the 7th IASTED*

International Conference on Visualization, Imaging and Image Processing, VIIP '09, pages 41–46. Acta Press, 2009.

[73] R. Lizarraga-Morales, R. Sanchez-Yanez, and V. Ayala-Ramirez. Visual texture classification using fuzzy inference. In I. Batyrshin and G. Sidorov, editors, *Proc. Tenth Mexican International Conference on Artificial Intelligence MICAI 2011*, pages 150–154, 2011.

[74] R. A. Lizarraga-Morales, R. E. Sanchez-Yanez, and V. Ayala-Ramirez. Fast texel size estimation in visual texture using homogeneity cues. *Pattern Recognition Letters*, 34(1):414–422, 2013.

[75] S. P. Lloyd. Least squares quantization. *IEEE Transactions on Information Theory*, 28(2):129–137, March 1982.

[76] A. Lobay and D. Forsyth. Recovering shape and irradiance maps from rich dense texton fields. In *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition,*, pages 400–406, Los Alamitos, CA, USA, 2004. IEEE Computer Society.

[77] A. Lobay and D. A. Forsyth. Shape from texture without boundaries. *International Journal of Computer Vision*, 67(1):71–91, Apr. 2006.

[78] L. Lucchese and S. Mitra. Colour image segmentation: A state-of-the-art. In *Proc. of the Indian National Science Academy*, volume 67, pages 207–221, March 2001.

[79] J. Maeda, A. Kawano, S. Yamauchi, Y. Suzuki, A. R. S. Marcal, and T. Mendonca. Perceptual image segmentation using fuzzy-based hierarchical algorithm and its application to dermoscopy images. In *IEEE Conference on Soft Computing in Industrial Applications, 2008. SMCia '08*, pages 66–71, 2008.

[80] L. Magdalena. What is soft computing? revisiting possible answers. *International Journal of Computational Intelligence Systems*, 3(2):148–159, 2010.

[81] P. Maji and S. K. Pal. Rough set based generalized fuzzy c-means algorithm and quantitative indices. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 37(6):1529–1540, 2007.

[82] J. Malik, S. Belongie, J. Shi, and T. Leung. Textons, contours and regions: Cue integration in image segmentation. In *Proc. of the International Conference on Computer Vision*, volume 2 of *ICCV '99*, pages 918–925, 1999.

[83] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th International Conference on Computer Vision*, volume 2, pages 416–423, July 2001.

[84] J. Melendez, M. A. Garcia, D. Puig, and M. Petrou. Unsupervised texture-based image segmentation through pattern discovery. *Computer Vision and Image Understanding*, 115(8):1121–1133, 2011.

[85] G. Menegaz, A. Franceschetti, and A. Mecocci. Fully automatic perceptual modeling of near regular textures. In *SPIE Human Vision and Electronic Imaging XII*, volume 6492, pages 64921B.1–64921B.12. SPIE, 2007.

[86] C. Meurie, A. Cohen, and Y. Ruichek. An efficient combination of texture and color information for watershed segmentation. In *Image and Signal Processing*, pages 147–156. Springer, 2010.

[87] M. Mignotte. Segmentation by fusion of histogram-based k-means clusters in different color spaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):780–787, May 2008.

[88] M. Mignotte. MDS-based segmentation model for the fusion of contour and texture cues in natural images. *Computer Vision and Image Understanding*, 116:981–990, 2012.

[89] A. Mohabey and A. Ray. Rough set theory based segmentation of color images. In *Proc. of the 19th International Conference of the North American Fuzzy Information Processing Society*, pages 338 –342, 2000.

[90] A. Mrozek and L. Plonka. Rough sets in image analysis. *Foundations of Computing and Decision Sciences*, 18(3-4):268–273, 1993.

[91] M. M. Mushrif and A. K. Ray. Color image segmentation: Rough-set theoretic approach. *Pattern Recognition Letters*, 29(4):483–493, 2008.

[92] M. M. Mushrif and A. K. Ray. A-IFS histon based multithresholding algorithm for color image segmentation. *IEEE Signal Processing Letters*, 16(3):168–171, 2009.

[93] P. Nammalwar, O. Ghita, and P. F. Whelan. Integration of feature distributions for color texture segmentation. In *Proc. of the 17th International Conference on Pattern Recognition*, volume 1, 2004.

[94] H. Ngan and G. Pang. Regularity analysis for patterned texture inspection. *IEEE Transactions on Automation Science and Engineering*, 6(1):131–144, jan. 2009.

[95] T. Ojala and M. Pietik˙ Unsupervised texture segmentation using feature distributions. *Pattern Recognition*, 32(3):477 – 486, 1999.

[96] T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[97] W. S. Ooi and C. P. Lim. Fuzzy clustering of color and texture features for image segmentation: a study on satellite image retrieval. *Journal of Intelligent and Fuzzy Systems*, 17(3):297–311, 2006.

[98] N. R. Pal and S. K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277–1294, 1993.

[99] S. K. Pal, B. Uma Shankar, and P. Mitra. Granular computing, rough entropy and object extraction. *Pattern Recognition Letters*, 26(16):2509–2517, 2005.

[100] M. Paulinas and A. Ušinskas. A survey of genetic algorithms applications for image enhancement and segmentation. *Information Technology and control*, 36(3):278–284, 2007.

[101] Z. Pawlak. Rough sets. *International Journal of Computer and Information Sciences*, 11(5):341–356, 1982.

[102] Z. Pawlak. *Rough Sets- Theoretical Aspects of Reasoning About Data*. Kluwer Academic Publishers, 1991.

[103] R. W. Picard, I. M. Elfadel, and A. P. Pentland. Markov/gibbs texture modeling: aura matrices and temperature effects. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 371–377. IEEE, 1991.

[104] M. Pietikainen and A. Rosenfeld. Image segmentation by texture using pyramid node linking. *IEEE Transactions on Systems man and Cybernetics*, 11:822–825, 1981.

[105] F. Precioso, M. Barlaud, T. Blu, and M. Unser. Robust real-time segmentation of images and videos using a smooth-spline snake-based algorithm. *IEEE Transactions on Image Processing*, 14(7):910–924, 2005.

[106] X. Qing, Y. Jie, and D. Siyi. Texture segmentation using lbp embedded region competition. *Electronic Letters on Computer Vision and Image Analysis*, 5(1):41–47, 2005.

[107] T. Reed and J. Dubuf. A review of recent texture segmentation and feature extraction techniques. *CVGIP: Image Understanding*, 57(3):359 – 372, 1993.

[108] F. Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386, 1958.

[109] A. Rosenfeld and E. B. Troy. Visual texture analysis. Technical report, Maryland Univ., College Park (USA). Computer Science Center, 1970.

[110] T. Saarela and M. S. Landy. Combination of texture and color cues in visual segmentation. *Vision Research*, 58:59–67, 2012.

[111] R. E. Sánchez-Yánez, E. V. Kurmyshev, and F. J. Cuevas. A framework for texture classification using the coordinated clusters representation. *Pattern Recognition Letters*, 24(1-3):21–31, 2003.

[112] M. Savelonas, D. K. Iakovidis, and D. Maroulis. An LBP-Based active contour algorithm for unsupervised texture segmentation. In *18th International Conference on Pattern Recognition, 2006. ICPR 2006.*, volume 2, pages 279–282, 2006.

[113] M. A. Savelonas, D. K. Iakovidis, and D. Maroulis. LBP-guided active contours. *Pattern Recognition Letters*, 29(9):1404–1415, 2008.

[114] G. Scarpa, R. Gaetano, M. Haindl, and J. Zerubia. Hierarchical multiple markov chain model for unsupervised texture segmentation. *IEEE Transactions on Image Processing*, 18(8):1830–1843, 2009.

[115] N. Senthilkumaran and R. Rajesh. Image segmentation - A survey of soft computing approaches. In *Proc. of the International Conference on Advances in Recent Technologies in Communication and Computing*, pages 844–846, 2009.

[116] N. Senthilkumaran and R. Rajesh. A study on rough set theory for medical image segmentation. *International Journal of Recent Trends in Engineering*, 2(2):236–238, 2009.

[117] L. Shafarenko, H. Petrou, and J. Kittler. Histogram-based segmentation in a perceptually uniform color space. *IEEE Transactions on Image Processing,*, 7(9):1354–1358, 1998.

[118] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, August 2000.

[119] J. Sklansky. Image segmentation and feature extraction. *IEEE Transactions on Systems, Man and Cybernetics*, 8(4):237–247, 1978.

[120] A. Skowron and L. Polkowski. Analytical morphology: mathematical morphology of decision tables. *Fundamenta Informaticae*, 27(2):255–271, 1996.

[121] T. Smith and J. Guild. The C.I.E. colorimetric standards and their use. *Transactions of the Optical Society*, 33(3):73–134, 1931.

[122] M. Spann and R. Wilson. A quad-tree approach to image segmentation which combines statistical and spatial information. *Pattern Recognition*, 18(34):257 – 269, 1985.

[123] H. Tamura, S. Mori, and T. Yamawaki. Textural features corresponding to visual perception. *IEEE Transactions on Systems, Man and Cybernetics*, 8(6):460 –473, june 1978.

[124] K. S. Tan and N. A. M. Isa. Color image segmentation using histogram thresholding fuzzy c-means hybrid approach. *Pattern Recognition*, 44(1):1 – 15, 2011.

[125] K. Thangavel, M. Karnan, and A. Pethalakshmi. Performance analysis of rough reduct algorithms in mammogram. *International Journal on Global Vision and Image Processing*, 5(8):13–21, 2005.

[126] C. W. Therrien. An estimation-theoretic approach to terrain image segmentation. *Computer Vision, Graphics, and Image Processing*, 22(3):313–326, 1983.

[127] S. Todorovic and N. Ahuja. Texel-based texture segmentation. In *IEEE 12th International Conference on Computer Vision*, pages 841–848. IEEE, 2009.

[128] S. Todorovic and N. Ahuja. Texel-based texture segmentation. In *Proc. of the International Conference on Computer Vision*, ICCV '09, pages 841–848, 2009.

[129] F. Tomita and S.Tsuji. *Computer Analysis of Visual Textures*. Kluwer Academic Publishers, 1990.

[130] M. Tuceryan and A. K. Jain. *The Handbook of Pattern Recognition and Computer Vision*, chapter Texture Analysis, pages 207–248. World Scientific Publishing, 1998.

[131] M. R. Turner. Texture discrimination by gabor functions. *Biological Cybernetics*, 55(2-3):71–82, 1986.

[132] R. Unnikrishnan, C. Pantofaru, and M. Hebert. A measure for objective evaluation of image segmentation algorithms. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 3, pages 34–41, 2005.

[133] R. Unnikrishnan, C. Pantofaru, and M. Hebert. Toward objective evaluation of image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):929–944, June 2007.

[134] M. Unser. Sum and difference histograms for texture classification. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(1):118–125, 1986.

[135] L. Van Gool, P. Dewaele, and A. Oosterlinck. Texture analysis anno 1983. *Computer Vision, Graphics, and Image Processing*, 29(3):336–357, 1985.

[136] S. R. Vantaram and E. Saber. Survey of contemporary trends in color image segmentation. *Journal of Electronic Imaging*, 21(4):040901–1–040901–28, 2012.

[137] S.-Y. Wan and W. Higgins. Symmetric region growing. *IEEE Transactions on Image Processing*, 12(9):1007–1015, 2003.

[138] A. Y. Yang, J. Wright, S. Sastry, and Y. Ma. Unsupervised segmentation of natural images via lossy data compression. *Computer Vision and Image Understanding*, 110(2):212–225, 2007.

[139] X. Yue, D. Miao, Y. Chen, and H. Chen. Roughness approach to color image segmentation through smoothing local difference. In *Rough Sets and Knowledge Technology*, volume 6954 of *Lecture Notes in Computer Science*, pages 434–439. Springer Berlin Heidelberg, 2011.

[140] L. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[141] J. Zhang and T. Tan. Brief review of invariant texture analysis methods. *Pattern Recognition*, 35:735–747, 2002.

[142] G. Zhao and M. Pietikäinen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):915–928, 2007.

[143] S.-C. Zhu, C.-E. Guo, Y. Wang, and Z. Xu. What are textons? *International Journal of Computer Vision*, 62(1-2):121–143, Apr. 2005.

[144] M. Zorman, P. Kokol, M. Lenic, J. Sanchez de la Rosa, J. F. Sigut, and S. Alayon. Symbol-based machine learning approach for supervised segmentation of follicular lymphoma images. In *Proc. of the IEEE International Symposium on Computer-Based Medical Systems*, pages 115–120. IEEE, 2007.

List of Abbreviations

CI : Computational Intelligence (See Chapter 1).

BDE : Boundary Displacement Error [34].

BSD : Berkeley Segmentation Data base and Benchmark [83].

GCE : Global Consistency Error [83].

NCuts : Normalized cuts method [118].

PRI : Probabilistic Rand Index [132].

PRM : Perceptual Roughness index-based segmentation Method (See Chapter 3).

RCT : Roughness index-based segmentation approach using Color and Texture features (See Chapter 5).

RBM : Roughness index-based segmentation Method [91]

RSLD : Roughness approach through Smoothing Local Difference [139].