مخبر الاتصالات، الاشارات والانظمه
Laboratory of Telecommunications, Signals and Systems

People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University Amar Telidji of Laghouat

Faculty of Technology

Department of Electronics

A Dissertation submitted to the faculty of technology in fulfillment of the requirements for the degree of Doctorate of Science

Major: Electrical Engineering

By

ZITOUNI ABDELKADER

Image Processing Methodology and Textures Analysis for their Segmentation

Méthodologie de Traitement d'Image et Analyse de Textures en vue de leur Segmentation

Presented and defended publicly in front of the jury:

Dr. Djerfaf Fatima	MCA at UATL	President of the jury
Pr. Chouireb Fatima	Professor at UATL	Supervisor
Dr. Benkouider Fatiha	MCB at UATL	Co-supervisor
Pr. Kious Mecheri	Professor at UATL	Examiner
Dr. Ahfir Maamar	MCA at UYF, Médéa	Examiner
Pr. Saadi Slami	Professor at UZA, Djelfa	Examiner

2019-2020

ABSTRACT

This thesis presents supervised classification algorithms based on information fusion for textured-images segmentation. Gabor features are efficient in finding class boundaries, whereas GLCM features are favorable in the areas within the classes. Moreover, the wavelets can represent textures at different scales and offer great discriminatory power between textures with strong resemblances. This motivates us to combine these three kinds of features to improve image segmentation. In the first step, the proposed method applied those three feature extraction strategies on textured images in order to get more information. After that in the second step, estimated feature vector of each pixel is sent to the neural networks classifier for pre-labeling. Then, in the third step of the proposed method, a classifier fusion method used to combine the scores obtained by the neural networks for each pixel. Finally, in the last step, in order to obtain more precise segmentation results, the scores within a sliding window are combined. The performance of the proposed segmentation algorithms was evaluated on synthetic images from Brodatz and DTD datasets and some MIAS images to detect masses, that represent one of breast cancer signs. The obtained classification results by the proposed fusions system leads to higher classification precision compared to applying a single classifier on the textured images.

Keywords: Image segmentation, Classification, Texture features, Gabor filters, Wavelets, GLCM, Information fusion.

ملخص

تقدم هذه الاطروحة خوارزميات تصنيف خاضعة للإشراف تعتمد على تقنية دمج المعلومات لتجزئة الصور ذات الصبغة النسيجية.

الخصائص المحصل عليها بعد استعمال مرشحات Gabor تتميز بالكفاءة في العثور على حدود المناطق، في حين أن الخصائص المحصل عليها باستعمال مصفوفات GLCM تتميز بالكفاءة في العثور على المساحات داخل المناطق المختلفة .علاوة على ذلك، يمكن لمحولة الموجات أن تمثل المناطق النسيجًية بمقاييس مختلفة وتوفر قوة تمييز كبيرة بين المناطق ذات التشابهات القوية .وهذا مما حفزنا على الجمع بين هذه الأنواع الثلاثة من الميزات لتحسين تجزئة الصورة. في الخطوة الأولى، تهدف الطريقة المقترحة الى تطبيق تقنيات استخراج الخصائص المذكورة اعلاه على الصور النسيجية للحصول على مزيد من المعلومات. بعد ذلك في الخطوة الثانية، يتم إرسال مجموعة الخصائص المستخرجة لكل بكسل إلى مصنف الشبكات العصبية من اجل الحصول على تقسيم اولي. ثم في الخطوة الثالثة من الطريقة المقترحة، تستخدم طريقة تصنيف عن طريق الدمج وذلك بدمج النتائج المحصل عليها من الشبكات العصبية لكل بكسل. أخيرًا، في الخطوة الأخيرة، ومن أجل الحصول على نتائج تجزئة أكثر دقة، يتم داخل كل نافذة منزلقة دمج كل النتائج المحصل عليها من الشبكات العصبية.

تم تقييم أداء خوارزميات التجزئة المقترحة على صور مركبة من مجموعة البيانات 'Brodatz' و'DTD' وبعض الصور لمجموعة البيانات MIAS وذلك للكشف عن الكتل، والتي تمثل واحدة من علامات سرطان الثدي. نتائج التصنيف التي تم الحصول عليها من قبل تقنيات الدمج المقترحة اظهرت دقة تصنيف عالية مقارنة بتطبيق مصنف واحد على الصور النسيجية.

الكلمات المفتاحية: تجزئة الصور، التصنيف، الخصائص النسيجية، مرشحات Gabor، محولة الموجات، مصفوفات GLCM، دمج المعلومات.

إهداء

الحمد لله الذي خلق فعلّم، وجعل بعد العسر يسراً، وبعد الجدّ والعمل ثمرةً ونجاحاً، والصلاةُ والسلامُ على الرّحمة المهداة نبينا محمدٌ صلى الله عليه وعلى آله وصحبه ومن والاه، أمّا بعد:

فأُهدي هذا العمل المتواضع إلى والديَّ العزيزين وإلى كلّ عائلتي وأحبائي وإلى جميع طلاب العلم، وأسألهُ سبحانه أن يرزقنا الاتقان في العمل وأن يجعلهُ خالصاً لوجه الكريم.

العلمُ تاجٌ ونبراسٌ لصاحبه يرقى به سلّم الأمجاد والرّتب
وان خلا المرءُ من علم ومعرفة فما يفيد جمال الوجه واللّقب

ACKNOWLEDGMENTS

Thanks and praises be to Allah the Almighty who gave me health strength and patience to complete this research study.

I would like to express my special appreciation and thanks to my supervisors Pr. Fatima Chouireb and Dr. Fatiha Benkouider. I would like to thank them for encouraging my research and for facilitating the way to accomplish my thesis. I would also like to thank my committee members, Doctor Djerfaf Fatima, Professor Kious Mecheri, Doctor Ahfir Maamar and Professor Saadi Slami for honoring me by accepting to be members in the jury of my thesis and help reviewing and improving this work. I would especially like to thank everyone helps me to get this work accomplished.

I would also like to thank my family. Words cannot express how grateful I am to my Parents for all of the sacrifices that they have made on my behalf. Their prayer and support for me were what sustained me thus far. I would also like to thank all of my friends who supported me in preparing my thesis, and incented me to strive towards my goal.

It is imperative also to mention the English professors Dr. Reggab Mourad, Pr. Belkheiri Mohamed, Dr. Mouissa Fattoum for their help in improving the English of my research papers and this thesis.

So that those that I have unintentionally forgotten and who participated in any way in the preparation of this work, find here the expression of my gratitude.

ZITOUNI Abdelkader

Laghouat University

February 2020.

LISTE OF ABBREVIATIONS

AR	Auto Regressive model.
bel(A)	Belief function of element A.
BPA	Basic Probability Assignment.
DT-CWT	Complex Wavelet Transform q-shift with Dual-Tree.
DTD	Describable Textures Dataset.
DWT	Discrete Wavelets Transform.
FN	False Negative.
FP	False Positive.
GAR	Graph of Region Adjacency.
GGD	Generalized Gaussian Distribution.
GLCM	Gray Level Co-occurrence Matrix.
HP	High Pass filter.
KNN	K-Nearest Neighbor.
LBP	Local Binary Pattern Method.
LP	Low Pass filter.
MAP	Maximum A Posteriori.
MIAS	Mammographic Image Analysis Society.
MLO	Medio-Lateral Oblique Mammograms.
MLP	Multi-Layer Perceptron.
Pl(A)	Plausibility function of element A.
RF1	Proportions of small ranges.
RF2	Proportions of large ranges.
SLP	Number of range length.
SVM	Support Vectors Machines
T-conorms	Triangular conorms.
TN	True Negative.
T-norm	Triangular norms.
TP	True Positive.
y_k^{des}	Desired output.

Table of contents

Table of Figures

Table of Tables

GENERAL INTRODUCTION

Introduction:

Image processing is part of a preliminary process to prepare images for their interpretation, storage or transmission after the acquisition and digitization steps. It is gaining increasing interest as the image becomes a support and a privileged source of information used in different areas, since the evolution of computer capabilities.

Image segmentation is an important step in any image analysis process. It is a low-level processing that precedes the measurement, comprehension and decision stage. Its aim is to divide the image into related and homogeneous regions in the sense of a homogeneity criterion that is difficult to define, particularly in the case of textured regions [1].

Several approaches have been proposed in the literature for image segmentation, Kennel et al. [2] proposed an algorithm for satellite images segmentation based on the complex wavelet transform method followed by Support Vectors Machines classifier for texton classification, Karine et al. [3] modeled the wavelet decomposition sub-bands by a generalized Gaussian distribution and used the KNN and Support Vectors Machines classifiers as measures of similarity for sonar images segmentation, Wu et al. [4] used the sub-bands energy of the scattering transform for texture discrimination followed by the Otsu thresholding method for features classification, to localize matriculation plaques, Wang and Zhang [5] presented a remote sensing image segmentation algorithm, which decompose the image into sub-band images using the wavelet transform to extract texture features, followed by the Bayesian classifier.

Texture, which is almost ubiquitous in images, is one of the most important features used to identify different regions of an image. Its study is difficult because there is no universal definition to apprehend the notion of texture. Texture segmentation and classification approaches have been used in literature and have shown their importance in various applications such as computer vision [6], image synthesis [7], biomedical image processing [8] and remote sensing [9-11]. Several methods have been proposed to address the problem of texture analysis, namely, the co-occurrence matrix [12], the range length matrix [13], the local bit patterns [14], Gabor filters [15], wavelet transform [16], scattering transform [17] ... etc. They can be divided into three main approaches: statistical, structural and spectral [18].

The purpose of texture analysis is to extract the characteristic properties of the object in an image and express them in a form of feature vector. The obtained representation will serve as a basis for later steps in image analysis and interpretation [19]. However, the search for discriminating parameters characterizing the texture and the use of these parameters for the segmentation still remains a delicate problem, without a universal solution. It is in this context that we suggest in this thesis new segmentation methods of textured images.

Two phases are usually involved in the texture classification process, namely the learning phase and the recognition phase. The learning phase aims to build a texture content model of each texture class present in the training data, which usually consists of images with known class labels. In the recognition phase, the texture content of the unknown sample is initially described using the same texture analysis method. Then, the textural features are compared to those of the training images with a classification algorithm, and the sample is assigned to the class with the best match [20]. The classification of these textural features can be done in a supervised way, which we know the membership classes of each pixel, or in an unsupervised way, which we have no knowledge a priori on the classes. For this, several types of classifiers exist, namely the K-nearest neighbor (KNN) [21], Support Vectors Machines [22], neural networks [23] ... etc.

Texture segmentation may be block wise or pixel wise. Texture features in the vicinity of each pixel within the image are evaluated in the pixel wise segmentation schemes. The benefit of pixel wise segmentation over block wise lies in the elimination of blockiness. Hence, we chose pixel wise segmentation for features extraction.

Generally, a single classifier is unable to handle the large variability and scalability of the data in any problem domain. Most modern techniques of pattern classification use a mixture of classifiers and combine the decisions offered to improve the classification precision. Feature fusion is of increasing importance for several computer vision-based applications including image classification [24]. As an example, to improve the classification precision there have been some proposed methods in the literature, like multi-modal learning [25-28], and multiple feature fusion [29-35].

The fusion of information consists of joining features from a number of sources to get better decision-making and it can be done at four levels depicted as follows [36]:

i. Sensor level fusion refers to the grouping of raw data from diverse sensors.
ii. Feature level fusion refers to the grouping of various features vectors.
iii. Score level fusion refers to the grouping of similar scores given by various classifiers.

iv. Decision level fusion refers to the grouping of decisions given by individual classifiers.

The fusion system is effective if the information from different sources is redundant and complementary. Redundancy can be exploited to reduce uncertainties and inaccuracies of information, and the complementarity can be used to obtain comprehensive and complete information and to remove ambiguities.

In the present research, new segmentation methods of textured images using three types of feature extraction techniques have been proposed. These features are statistical features taken out of GLCM matrices and structural features obtained using Gabor filters and the wavelet transform coefficients.

Combining three types of features increases the classification performance since it uses Gabor features which are more efficient in near-class borders areas combined with GLCM features which are powerful in the areas within classes, and the wavelet coefficients that offer great discriminatory power for textures with strong resemblances.

In the first step of our work, we applied these three features extraction strategies on textured image in order to get more information that enables the classifiers to discriminate the textures. In the second step, an appropriate classification algorithm is applied using the set of extracted features in the previous step. The neural networks classifier is chosen among the most well-known classifiers, it was initiated in [37, 38]. The estimated feature vector of each pixel is sent to the neural networks classifier for pre-labeling. For the training phase of the neural networks, images of 320×320 size were used, and 6400 patterns are selected randomly from each image, which represents 6.25% of the total number of patterns. In the third step of the proposed method, a Bayes fusion method is used for each pixel to combine the three scores obtained by the neural networks of each kind of features.

A sliding window, whose class is allocated to its central pixel is used. Nevertheless, this central pixel belongs to other window neighbors that may be classified into other classes. Therefore, in the last step, in order to obtain a more precise segmentation result, a Bayes fusion method is used for each pixel to combine the scores results of various windows that include this central pixel.

Once the classification is done, the system must be evaluated. In general, we rely on a visual evaluation (visual comparison of the automatic classification results). In this work, numerical measures are also used for the classification and segmentation evaluation, these measures are more important than a simple visual evaluation.

The performance of the proposed algorithms for segmenting textured images is assessed using many images with different textures and different shapes. They are collected in the wild and in cluttered conditions, from Brodatz and DTD datasets. The obtained results lead to higher classification precision compared to applying a single classifier on textured images. The rest of this thesis is organized as follows:

In the first chapter, we are interested in classical methods of texture analysis by classifying them according to four main families: geometric methods based on the extraction of the basic patterns of texture, methods for extracting descriptors by a spatial analysis luminous intensity, methods based on frequency or spatio-frequential analysis (wavelets).

The second chapter is devoted to the study of segmentation methods, particular the segmentation resulting from the classification. The different classification methods: k-nearest neighbors (KNN), K-means, neural networks, and the classification assessment approaches are presented.

The third chapter focuses on the information fusion. Starting by presenting the general notions of the fusion and the different theories used for this purpose, especially the theory of uncertainty.

In chapter 4 we present all contributions made in this framework, and we evaluate the performances of the proposed systems. The influence of fusion on the overall performances of the classification system are studied. Then, a computer-assisted diagnostic system for masses' detection in mammography images is proposed.

Before presenting the appendices, we draw all conclusions related to the different aspects of our work, which open up a set of perspectives on adjacent problems.

CHAPTER I

Images Search by Texture Content: State of Art

I.1. Introduction

Before beginning the content of the chapter, it is important to understand the nature of the objects we are going to manipulate. Computationally, an image is a digital representation in memory of a subject printed on an artificial retina (matrix as the sensor of a digital camera or the virtual scene of a synthetic image, or linear such as the optical sensor of Fax machine, photocopier or scanner). We will therefore work on sets of numerical numbers coded on computer.

Digital image processing will implement two main types of approaches:

- Enhancement of images to be visualized and eventually to be interpreted by a human expert.
- Computer vision which consists to make automatic perception's operations in a similar way to the human visual perception system.

In this chapter, we first introduce the basic notions and concepts to characterize the visual content of images based on texture, then we present the state of art of the most commonly used methods in texture analysis.

I.2. Texture Notions

I.2.1. Definition:

Texture is a fundamental feature of images and pattern recognition. However, despite its importance and its presence in most images, there is no precise and consensual definition of texture in computer vision, it is easily perceived by humans and a rich source of visual information.

In general, the texture can be defined as a regular repetition of an element or pattern on a surface. Rosenfeld [39] defined the texture as a grouping of similarity in an image, Haralick [40] described a textured image as an organization phenomenon of a region defined by the number and type of its primitives as well as their spatial organization or arrangement. It has two dimensions for its description: the first concerns the primitives of which it is composed, the second allows to describe the spatial dependence or the interaction between primitives. This dependence can be structural, probabilistic or functional.

The performance of texture methods depends on the area on which they are applied and also the nature of texture in that area.

Let's analyze the images of figure I.1, we notice that these images are strongly textured: with zones presenting different types of texture. It is thus possible to characterize these zones by a texture analysis.

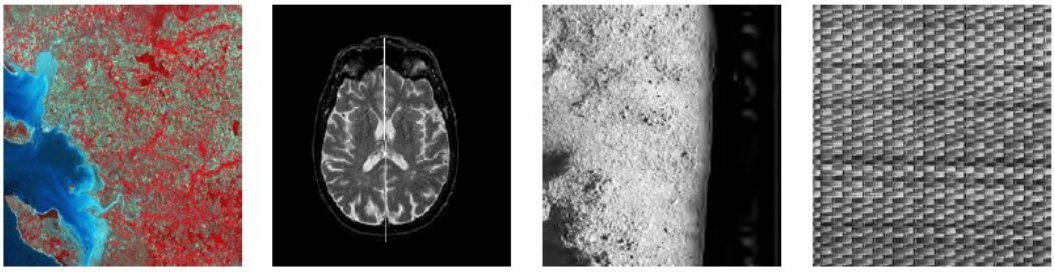


Figure I.1 Example of textured images: remote sensing, medical, sonar, synthetic.

I.2.2. Texture types

In the literature, the classification into textural families is often related to the definition used for texture. Two big families are needed: deterministic textures and stochastic textures (probabilistic) [41].

A texture is called deterministic (Macro texture) if it is formed by a regular arrangement of a geometric pattern. This type of texture is fully characterized by a pattern and its deterministic layout rule. This definition is suitable only for perfectly regular textures that we rarely meet in reality (See figure I.2).

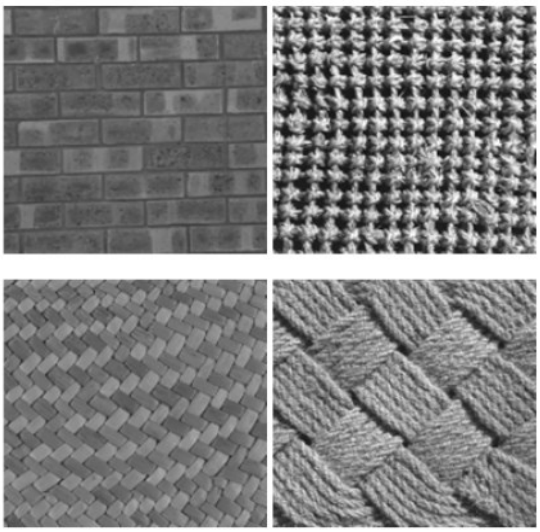


Figure I.2 Examples of deterministic textures

A texture is called stochastic (Micro texture) if it is characterized by an irregular content from which it is sometimes difficult to extract a basic primitive (non-locatable pattern). It is

described by laws or statistical models and is considered as a realization of a homogeneous two-dimensional random field (See figure I.3).

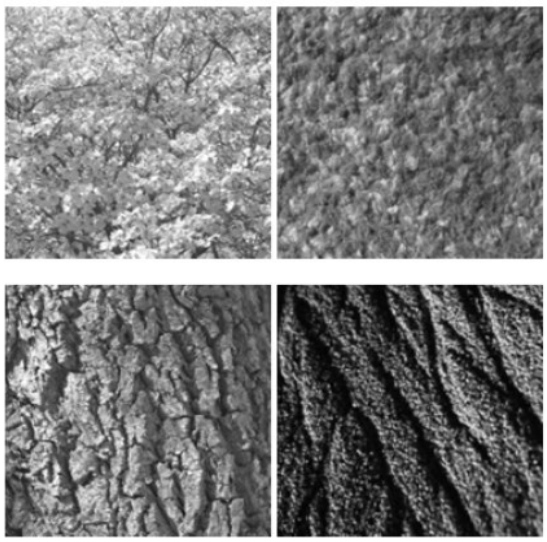


Figure I.3 Examples of stochastic textures

But it is sometimes difficult to classify a texture in one or other of the categories (mixed textures). This shows that the notion of texture is closely linked to visual perception.

I.3. Texture analysis

We cannot directly exploit the information contained in an image through the gray level of pixels, because this information is not sufficient to represent such complex images and does not allow to obtain realistic response times for an image search system. It is therefore appropriate to use a reduced-size representation to characterize an image, hence the need to extract textural information in order to discriminate between the different regions of the image.

Texture analysis is a crucial step in the classification of textures, image segmentation, shape recognition and compression of images. It gathers a set of quantification methods of the gray levels present in an image in terms of intensity and distribution in order to calculate a number of characteristic parameters of the texture in consideration. It is generally treated by two phases: the first is the extraction of image characteristics, the second is the classification of the obtained characteristics of pixels.

The study of texture has been the subject of several researches that have resulted in a multitude of texture analysis methods which can be divided into 4 categories [42].

I.3.1. Statistical approaches

Statistical methods consist of evaluating the statistical properties of a region or neighborhood around a pixel as textural information. It is obvious that these measurements depend on the size of the region or the analysis window. This type of method is adapted to stochastic textures (Micro texture).

Statistical methods can be ranked first-order, second-order, and higher-order based on the number of pixels that define the local characteristics. The higher the order of statistics is, the greater the number of pixels is involved.

I.3.1.1. First order statistics

They take into account one pixel at a time to estimate properties, ignoring the spatial interaction between pixels of the image, for example the mean, variance and the histogram ... etc.

- ***Mean***: Gives the average gray level in a window W containing N pixels:

$$\text{Mean} = \frac{1}{N}\sum_{i,j} W(i,j) \qquad I.1$$

- ***Histogram***: The intensity level histogram is a concise and simple method summarizing the statistical data contained in an image. The calculation of the gray level histogram involves individual pixels. Thus, histogram contains the first-order statistical information about the image.

Histogram is defined as a function that represents for each level of intensity, the number of pixels in the whole image, which have the same intensity:

$$h(j) = \sum_{x=0}^{l-1}\sum_{y=0}^{k-1} \delta(g(x,y),j) \qquad I.2$$

With: $\delta(i,j)$ the Kronecker symbol.

$$\delta(i,j) = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \qquad I.3$$

$g(x,y)$ is the function that represents the image in two-dimensional space, $x = 0 \ldots l-1$ and $y = 0 \ldots k-1$.

$j = 0 \ldots G-1$, with G is the total number of intensity levels in image.

Dividing $h(j)$ by the total number N of pixels in image, the approximate probability density of intensity levels is obtained.

$$p(j) = \frac{h(j)}{N} \qquad I.4$$

Obviously, several parameters can be extracted from the histogram such as mean, variance, kurtosis, energy, and entropy [42]. Although these statistics may be discriminating properties in simple cases, they remain a limited representation of textures compared to higher order statistics.

I.3.1.2. Second order statistics

They exploit pixel pairs by including some spatial dependence. Among the most popular second-order statistics for texture analysis is the co-occurrence matrix [12].

I.3.1.2.1. Gray Level Co-occurrence Matrix (GLCM):

Gray level co-occurrence matrix notion and features have been initiated in more than three decades ago. It is a powerful technique to extract texture features and yet many efforts are being made to enhance the algorithm's speed and performance, or mix it with new techniques [43].

It is a matrix G of size n_g x n_g calculated from an image I represented in n_g gray levels of size M x N, it permits to determine the observation frequency of a pattern formed of two pixels separated by a distance d in a particular orientation θ. Generally, $\theta = k\pi/4$, (k = 0, 1, 2, 3, 4) and d = 1, 2. The formula I.5 defines the cooccurrence matrix for an image I:

$$G_{d\cos\theta, d\sin\theta}(i,j) = \sum_{p=1}^{p=M}\sum_{q=1}^{q=N} \begin{cases} 1 & if\ I(p,q) = i \quad with \quad I = (p + d\cos\theta, q + d\sin\theta) = j \\ 0\ , & otherwise \end{cases} \qquad I.5$$

For each distance-direction pair *(d, θ),* a $GLCM_{d,\theta}$ matrix is calculated.

Let figures I.4 and I.5 respectively illustrate the pixels' arrangement for the co-occurrence matrix calculation and its calculation method, with $d = 1$ and θ = 0°, 45°, 90°, 135°, for an image I of size 3 x 4 with 6 gray levels.

A step of normalizing the co-occurrence matrix $G(i,j)$, resulting from the sum of several orientations / distances, is required before extracting texture parameters. The normalization is done by dividing the co-occurrence matrix on the sum of its elements.

$$G_N(i,j) = \frac{G(i,j)}{\sum_{i=1}^{n_g}\sum_{j=1}^{n_g} G(i,j)} \qquad I.6$$

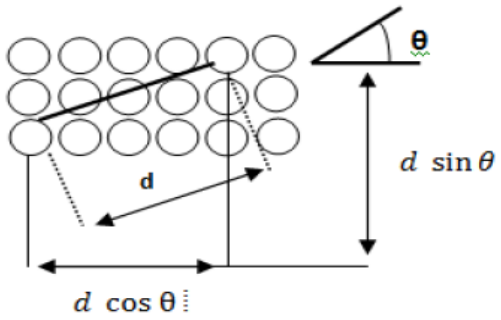


Figure I.4 Pixel arrangement for calculating the co-occurrence matrix [53].

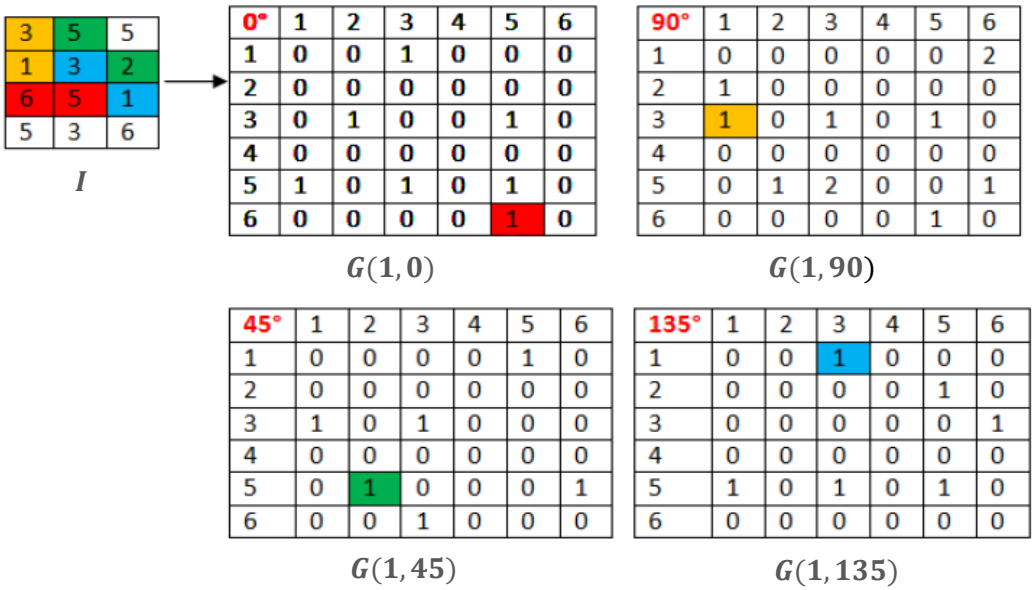


Figure I.5 Principle of the co-occurrence matrices computation ($\theta = 0°, 45°, 90°, 135°$ and $d = 1$) [53].

The GLCM features are very close to human inference from texture and they differentiate image texture very well; however, they require powerful processing resources. Several approaches are suggested to overcome this issue. The simplest one is reducing the gray level of the image through re-sampling, which reduces the dimensions of GLCM matrix [44].

The size of the neighborhood window is a major parameter in GLCM computations to be considered. Better discrimination in regions near borders are, theoretically, obtained by small window sizes, though, practically, they generate sparse GLCM matrices which give rise to imprecise feature extraction process. Alternatively, although large window sizes will give rise to more accurate extracted features, they cause diverse classes to merge in the regions close to the class borders. Thus, choosing suitable neighborhood window size is an essential step in GLCM process [43].

After obtaining GLCM matrices over the sliding window in the input image, several statistics like contrast, uniformity, and entropy will be extracted from these matrices [43]. Haralick suggests the calculation of fourteen features [45], not all relevant for a given application:

Homogeneity: It has a high value for uniform images and for periodic textured image in the direction θ [45], and it is given by:

$$\sum_{i=1}^{n_g}\sum_{j=1}^{n_g} G(i,j,d,\theta)^2 , \qquad I.7$$

where n_g *is* the number of gray levels and $G(i,j,d,\theta)$ is the estimation of the likelihood of transition from the pixel i to the pixel j with a distance d in the direction θ [45].

Contrast: It characterizes the big transition probabilities between pixels with high differences in gray level and it is given by:

$$\frac{1}{n_g-1}\sum_{k=0}^{n_g-1} K^2 \sum_{i,j=1:|i-j|=K}^{n_g} G(i,j,d,\theta) \qquad I.8$$

Entropy: It represents information present on the image:

$$1-\sum_{i=1}^{n_g}\sum_{j=1}^{n_g} G(i,j,d,\theta)\log\big(G(i,j,d,\theta)\big) \qquad I.9$$

Correlation between image rows and columns:

$$\sum_{i=1}^{n_g}\sum_{j=1}^{n_g}\frac{(i-\mu_x)(j-\mu_y)G(i,j,d,\theta)}{\sigma_x\,\sigma_y} , \qquad I.10$$

where μ_y and μ_x describe the mean on columns and rows of G successively, and σ_y and σ_x are the standard deviations.

The directivity:

$$\sum_{i=1}^{n_g} G(i,i,d,\theta) \qquad I.11$$

Texture with high direction privilege in the direction θ has a high directivity.

Uniformity: It gives the proportion of each gray level in the image:

$$\sum_{i=1}^{n_g} G(i,i,d,\theta)^2 \qquad I.12$$

The co-occurrence matrices are widely used in image analysis, we find applications such as the detection of different types of clouds present in a satellite image by [46], segmentation of old documents [47] and sonar images [48]. This method will be used in our study and will be compared with other approaches of texture analysis.

I.3.1.2. Higher order statistics

I.3.1.2.1. Range length matrix

This matrix contains information about the range, which is a set of consecutive pixels having the same gray level in a given direction [13]. The number of pixels in a given range represents the range length.

Each element of the range length matrix $M(i,j)$ represents the range number $n\theta$ of length j and gray level i, in a direction θ. Figure I.6 explains the calculation principle of the range length matrix for an image I of size 3 x 3 at 3 gray levels (GL) (0 - 2), in 4 directions (θ = 0°, 45°, 90°, 135°) like the cooccurrence matrix principle previously described.

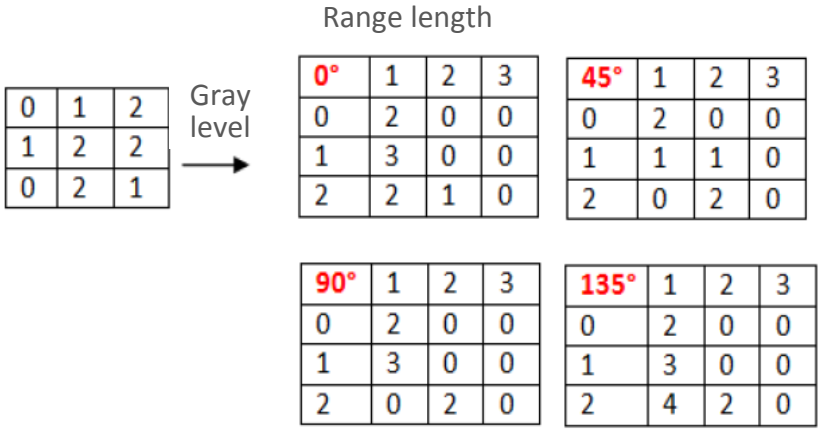


Figure I.6 Calculation of the range length matrix for θ = 0°, 45°, 90°, 135°.

Several statistical parameters can be calculated from the range-length matrices in a similar way to [40] for the co-occurrence matrices. Some measures are presented below to illustrate the characteristics' calculation through these matrices [49].

- ***Number of range length***

$$SLP = \sum_{i=0}^{GL} \sum_{j=1}^{n\theta} M_\theta(i,j) \qquad I.13$$

- Proportions of small ranges

$$RF1 = \frac{1}{SLP}\sum_{i=0}^{GL-1}\sum_{j=1}^{n\theta}\frac{M_\theta(i,j)}{j^2} \qquad I.14$$

- Proportions of large ranges

$$RF2 = \frac{1}{SLP}\sum_{i=0}^{GL-1}\sum_{j=1}^{n\theta} j^2\, M_\theta(i,j) \qquad I.15$$

This method has been the subject of several comparative studies of texture analysis methods[50,51].

I.3.2. Structural approaches

The structural methods represent texture by the description and the determination of rules conditioning the position of basic elements called primitives, then to describe texture, it is necessary to define the primitives and their rules of positioning. The choice of a primitive from a set of primitives and the probability that the primitive is positioned at a given location may be a function of the primitive's position or adjacent primitives to the position [12]. This kind of method is suitable for deterministic, regular textures.

I.3.2.1. Local Binary Pattern Method (LBP)

Ojala [14] introduced the local binary pattern method which is based on the comparison of the luminance level of a pixel with his neighbors' levels.

For each pixel of a gray-scale image, a neighborhood V of radius R, for example $V = V_0$, V_1 ... V_8, is defined. The gray level of neighboring pixels is compared with the central pixel gl_c in order to generate a binary pattern ($gl_0 - gl_c$, ..., $gl_8 - gl_c$). If $gl_i < gl_c$ with $i = 1\ldots 8$, we assign to the neighboring pixel i the value 0, otherwise we assign it the value 1. The binary values of this pattern are multiplied by weights and summed to calculate the LBP code of the central pixel (see figure I.7). The histogram of the LBP codes is the descriptor characterizing the texture. It is considered a unifying approach since it combines the statistical and structural aspect of texture analysis [52].

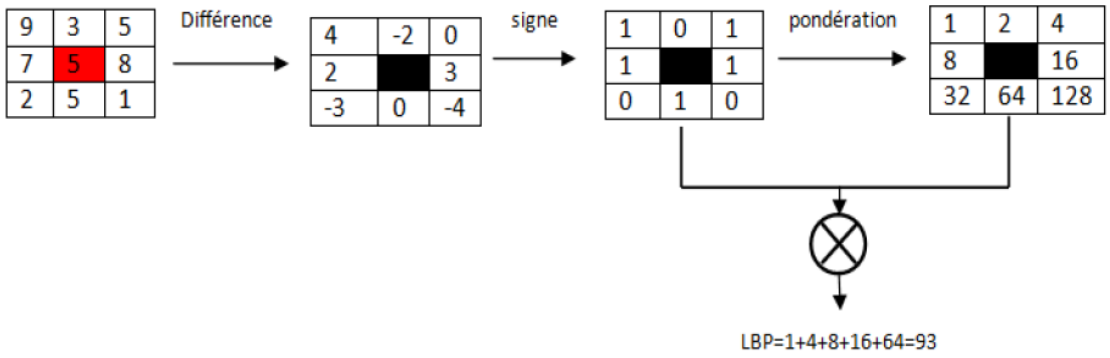


Figure I.7 Calculation principle of the LBP code.

I.3.3. Frequency approaches

The frequency methods, or so-called filtering methods, permit to analyze a texture by identifying the different frequencies that compound it. This kind of method is suitable for quasi-periodic signals, such as the case of textures.

I.3.3.1. Gabor filters

The Fourier transformation of an image makes it possible to highlight the regularities of the texture by examining the frequency domain. The problem posed by this operation, which acts globally on the image, is that it does not take into account the spatial location.

A proposed solution is to use an alternative transformation called a Fourier transformation with sliding window, which his principle is to apply the Fourier transformation in an observation window that moves on the image. The choice of the window size depends on the spatial characteristics of textures to be analyzed. There are different observation windows: triangular, hamming, and Gauss window. When the latter is applied, we talk about the Gabor transformation [53].

Gabor wavelets have been widely used in diverse fields of image processing like fingerprint identification, image coding, edge detection and texture classification [43]. They are utilized to extract image texture features. The idea is based on detecting linear directional elements in the image.

A Gabor wavelet is a linear filter whose impulse response is defined as a harmonic function multiplied by a Gaussian function. Gabor filter orientation and frequency representations are alike to those of human visual system [54]. In the spatial domain it is given by [55]:

$$h(x,y) = g(x,y) \times cos(2\pi(u_0\, x)), \qquad I.16$$

where $g(x,y)$ is a 2D Gaussian defined as:

$$g(x,y) = e^{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right]}, \qquad I.17$$

and u_0 is the spatial frequency of a cosine along the x axis and orientation of 0^0, and (σ_x and σ_y) are the spatial variances of the Gaussian envelope along the x and y axes, successively [55]. The remaining filters, with orientations of 0º, 45º, 90º and 135º are obtained by rotating the co-ordinate system, or by using the rotation equation for the (x,y) plane, where θ_0 represents the rotation angle [55]:

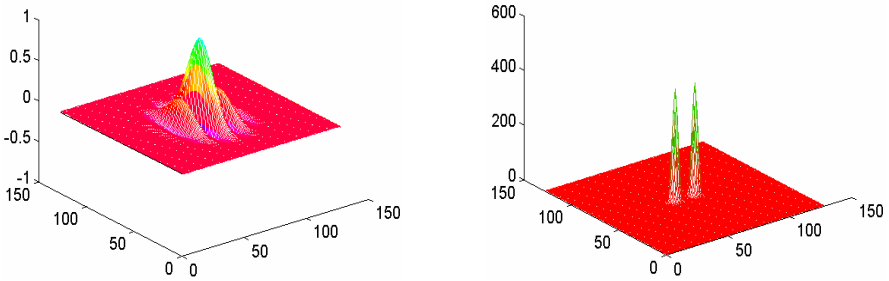
$$h(x,y) = e^{-\frac{1}{2}\left[\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right]} \cos\big(2\pi u_0(x\cos\theta_0 + y\sin\theta_0)\big) \qquad I.18$$

The orientation selectivity properties of the Gabor filter are far clearer in the frequency domain. So, the Fourier transform of equation $(I.18)$ is given by the subsequent expression [55]:

$$H(x,y) = A\left\{\exp\left(-\frac{1}{2}\left[\frac{(u-u_0)^2}{\sigma_u^2}+\frac{v^2}{\sigma_v^2}\right]\right) + \exp\left(-\frac{1}{2}\left[\frac{(u+u_0)^2}{\sigma_u^2}+\frac{v^2}{\sigma_v^2}\right]\right)\right\}, \qquad I.19$$

where $\sigma_u = {}^{1}/_{2\pi\sigma_x}$, $\sigma_v = {}^{1}/_{2\pi\sigma_y}$ and $A = 2\pi\sigma_x\sigma_y$.

Figure I.8 shows an even-symmetric Gabor filter and its frequency representation in a 64 × 64 array, and orientation of 0º.



a b

Figure I.8 Symmetric Gabor filter and its frequency representation.
(a) Spatial representation of the Gabor Filter, ***(b)*** Frequency representation of the Gabor Filter.

Texture segmentation requires simultaneous measurements in both the spatial and the spatial-frequency domains. Filters with smaller bandwidths in the spatial frequency domain are more desirable because they allow us to make finer distinctions among different textures. On the other hand, accurate localization of texture boundaries requires filters that are localized in the spatial domain. However, the effective width of a filter in the spatial domain and its bandwidth in the spatial frequency domain are inversely related. An important property of Gabor filters is that they have optimal joint localization, or resolution, in both the spatial and the spatial-frequency domains [56].

The parameters of the Gabor function are specified by the frequency, the orientation of the sinusoid (or represented by the center frequency), and the scale of the Gaussian function. Local orientations and spatial frequencies explicit in Gabor filters are therefore used as the key features for texture processing. The input image is generally filtered by a family of Gabor filters tuned to several resolutions and orientations.

So, to extract image texture features, an ensemble of wavelets is created using a mother wavelet like in equation ($I.18$). Then the entire image is used as the input to the wavelet set. By doing so, the input image is transformed into a set of N filtered images. As the filters output are complex valued, the magnitude of these values is generally exploited.

I.3.3.2. Wavelet analysis

Since the work of Grossman and Morlet [57], the wavelets transform has appeared as a powerful tool to solve problems in different application. The wavelets transform decomposes the input signal into a series of wavelet functions $\Psi_{a,b}(t)$ derived from a mother function $\Psi(t)$ given by dilatation (factor a) and translation (factor b) operations. Figure I.9 shows some examples of wavelets that are usually used in image processing [1].

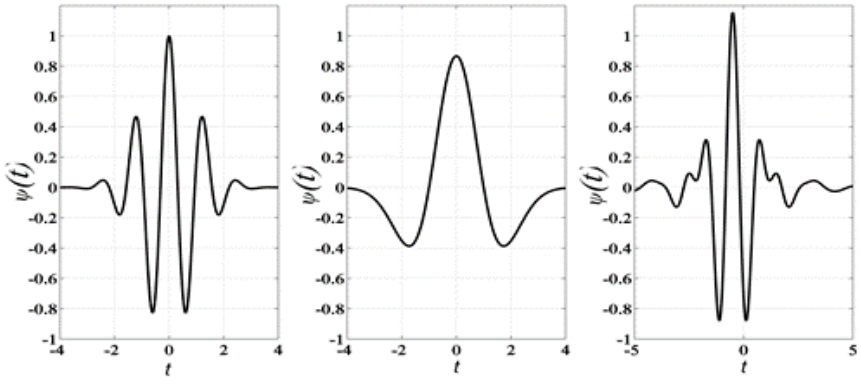


Figure I.9 Classical examples of wavelets.
(a) Wavelet of Morlet, ***(b)*** Wavelet of Mexican hat, ***(c)*** Wavelet of Meyer.

Wavelet analysis transforms a finite energy signal in the spatial domain into another finite energy signal in the spatio-frequency domain. The components $C_{a,b}$ of this new signal, described in equation $(I.20)$, are called wavelet coefficients. In an image, these coefficients offer information on the local variation of the gray levels around a given pixel. The more significant is this variation, the higher they are [18].

$$C_{a,b} = \int_{-\infty}^{+\infty} x(t)\, \Psi_{a,b}(t)dt, \qquad I.20$$

where

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{a}}\Psi\left(\frac{t-b}{a}\right) \quad \text{with} \quad a \neq 0 \qquad I.21$$

The major advantage of wavelets as compared to other frequency methods, as Fourier transform, is that it offers both frequency and spatial locality [20].

In 1989, Mallat [16] proposed a multi-resolution decomposition algorithm based on wavelets transform. The algorithm decomposes an input image into a set of detail images and an approximation image using a filter bank comprising a high pass filter (HP) and a low pass filter (LP). At each decomposition level the size of the transformed images is reduced by a factor of two [18]. The discrete wavelets transform of a 2-D image can be obtained by performing the filtering successively along horizontal and vertical directions (separable filter bank) [58]. Four images are then generated at each level.

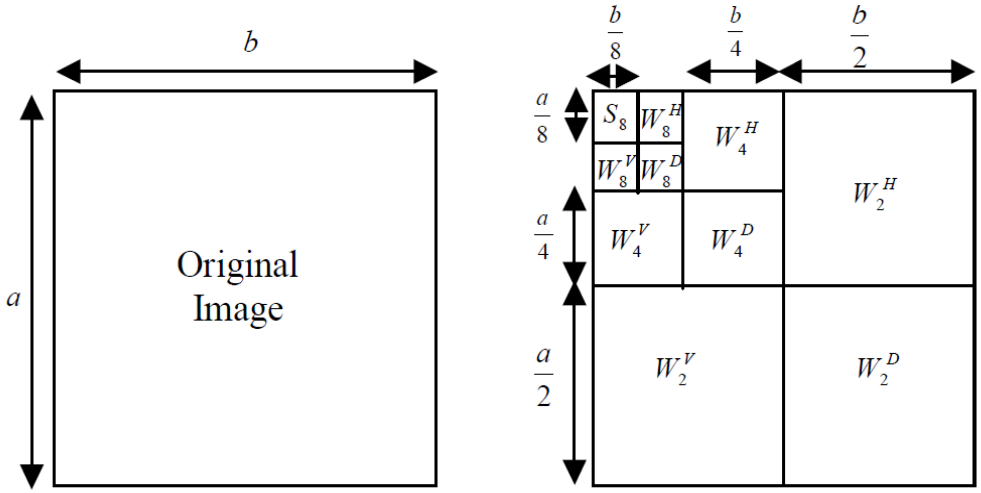


Figure I.10 Pyramidal discrete wavelets transform (DWT).

Figure I.10 shows a three-level pyramidal wavelet decomposition of an image of size a x b pixel. In the first level of decomposition, one low pass sub-band image LL_1 (contains vertical

and horizontal low frequencies, it represents approximation coefficients) and three orientation selective high pass sub-band images: HL_1(contains vertical low and horizontal high frequencies, it represents horizontal details), LH_1(contains vertical high and horizontal low frequencies, it represents vertical details) and HH_1(contains vertical and horizontal high frequencies, it represents diagonal details) are created. In second level of decomposition, the low pass sub-band image is further decomposed into one low pass and three high pass sub-band images (LL_2, HL_2, LH_2, HH_2). The process is repeated on the low pass sub-band image to form higher level of wavelet decomposition. In other words, DWT decomposes an image in to a pyramid structure of the sub images with various resolutions corresponding to the different scales. We note that a three-stage decomposition will create three low pass sub-band images and nine (three in each horizontal, vertical, and diagonal direction) high pass directional sub-band images. The low pass sub-band images are low-resolution versions of the original image at different scales. The horizontal, vertical and diagonal sub images provide the information about the changes in the corresponding directions respectively.

The values of transformed coefficients in detail and approximation images (sub-band images) represent the necessary features that capture useful discrimination information for texture segmentation [59].

The wavelets transform has a variant known as a "wavelets packet transform" which allows a very fine analysis of textures. At each level, the sub-bands are selected for the next decomposition. The difference between the wavelets transform and the wavelets packet transform is that this last wavelets transform decomposes the high frequency components thus building a multi-scale tree extension of the wavelets transform [1].

- ***Calculation of the texture indices***: One of the most used indices for characterizing the texture in the spatio-frequency plane is the measurement of energy. Because the transformed images have different frequencies, orientations and scales, the energy index is a local measure of the wavelet coefficient distribution according to the frequency, the orientation and the scale. It has been used successfully for classification and segmentation of textures [18].

  The expression of energy expressed in standardized form is given by [18]:

$$E = \frac{1}{N}\sum_{R} C(i,j)^2 \qquad I.22$$

  The second index used in conjunction with energy is the measurement of the local mean of the wavelet coefficients given by [18]:

$$M = \frac{1}{N}\sum_{R}|C(i,j)|, \qquad I.23$$

where N denotes the number of pixels, named by the indices (i,j), and enclosed in the neighborhood R on which the texture is estimated.

The evaluation of these indices is done on a sliding window W. The energy and the local mean on the sliding window are calculated from the resultant sub-band images. So, if a wavelet on one level is used, the feature vector of each window is made of eight parameters V= [E_{LL}, E_{LH}, E_{HL}, E_{HH}, M_{LL}, M_{LH}, M_{HL}, M_{HH}].

This method has been widely used in the literature for the segmentation of textured images [5]. In [60] the authors used the characteristics of the co-occurrence matrix derived from the wavelets transform on blocks of the image.

I.3.4. Approaches by modelling

These approaches define texture as the realization of a stationary random process and characterize it using fractal and stochastic models. The description of the texture properties is then reduced to the identification or estimation of the model's parameters. In practice, the first problem is the computation's complexity for estimating the stochastic model. The fractal model can be used for the analysis of natural textures. It can be used for texture analysis as well as for texture discrimination. However, it lacks the orientation's selectivity and is not able to describe the local structures of an image.

Among the most used models are the Markov models whose basic idea is that the knowledge of the neighborhood V of a pixel is sufficient to calculate its conditional law. In the case of Markovian Gaussian models, the autoregressive model (AR) considers each central pixel in a neighborhood as a weighted sum of neighbors and an independent additive noise identically distributed in order to characterize the local interactions between pixels.

Two classes of probabilistic models are possible to distinguish the distributions of the sub-band coefficients resulting from a multi-resolution representation [41]:

- The univariate models describing the marginal statistics of the sub-bands which are considered as the realization of a random variable (scalar). These models are a parameterization of the sub-band histogram method. The univariate model most widely used is the generalized Gaussian distribution (GGD).
- The multivariate models that generalize the precedents by describing the joined statistics of the sub-bands which are considered as the realization of a random field (vector). These models

are a parameterization of the multidimensional histograms of the sub-bands and allow to capture the dependence between coefficients.

For our study, the co-occurrence matrix, wavelet transform and Gabor filters will be used.

I.4. Conclusion

In this chapter, the texture notions and the state of art of the most used and most recent texture analysis techniques are presented. In the next chapter we present the different segmentation methods dedicated to textured images including method based on pixel classification, giving the main classification methods, namely k-nearest neighbor, neural networks… etc.

CHAPTER II

Textured Images Segmentation

II.1. Introduction

Segmentation is a low-level processing that precedes the measurement, understanding and decision stage. It is an essential step in image analysis and pattern recognition process, whose quality of which influences the success or failure of the system. It consists of analyzing and cutting out an image in uniform zones according to some criteria such as color, texture or gray levels, where the regions will be labeled later. Within the framework of this first study of textured images segmentation, the criterion retained is texture.

Segmentation is easily done by humans but remains one of the most active research areas in image processing. A large number of segmentation methods have been studied in the literature, namely segmentation by the watershed [61], by Markov chain [62], by probabilistic techniques [63], and by complex wavelet transform q-shift with dual-tree (DT-CWT) and SVM [2].

This chapter introduces the segmentation methods suitable for textured images, especially the region-based segmentation methods because in general a texture does not contain any clear boundaries, so the contour-based methods will not be detailed. The study will be based on segmentation founded on pixel classification.

The success of the classification stage depends on the nature of data and the classifier used. We give an overview of the most used classifiers in the state of art, namely the k-nearest neighbors, k-means and MLP.

II.2. Segmentation methods of textured images

Segmentation methods can be grouped into three families: contours, regions and classification. These methods are called locally methods, because they work at the pixel level and not the overall image.

II.2.1. Contour Detection Approaches

These approaches consist in detecting the variation between two image regions. The pixel gray level is the only information taken into account to perform this task. These techniques are generally too limited to deal with noisy and complex images and are more suitable for uniform images.

The edge detection methods can be divided into two families [64]:

- Derivative methods.
- Methods by variational approach.

II.2.1.1. Derivative methods

This category is based on the calculation of the order 1 or 2 derivative for the contour detection:

- ***Operator of order 1***: Also called gradient approach because the determination of local extremum is based on the gradient direction. Among these operators are Sobel, Prewit, Canny, Dériche ...
- ***Operator of order 2***: Also called the Laplacian approach because the location of contours is based on the determination of zero crossings of the Laplacian.

II.2.1.2. Methods by variational approach

In this category we find the model of active contours (snake). These contours are represented by curves which deform from an initial position belonging or close to the interest object [65]. This displacement and deformation are based on the concept of internal and external energy, the aim of which is to minimize the total energy present along the curve. For more details see [66].

II.2.2. Approaches by regions

These approaches group together a set of connected pixels with similar characteristics in order to constitute homogeneous and coherent regions. They favor the spatial interaction between pixels. We can distinguish four categories:

- Regions' Growth.
- Regions' Divisions.
- Regions' Fusions.
- Decomposition / Fusion.

II.2.2.1. Regions' Growth

Originally introduced by [67]. These methods start from pixels of the image, selected in a random or deterministic way and group them in an iterative manner according to a homogeneity criterion. The growth stops when you can no longer add pixels without breaking the homogeneity. The most frequent way consists of starting by pixels on the top left of the image to go towards the down right of image.

The advantage of this method is to keep the shape of each region of the image, as well as its speed and ease of implementation. However, a bad selection of the initial pixels or the poorly adapted choice of the homogeneity criterion can cause problems of sub/over-segmentation. Therefore, it is advisable to choose the initial points in the most homogeneous areas possible.

This method has been widely used in the literature for image segmentation; [68,69] proposed a textured images segmentation method by region growth, Lira et al. [70] proposed a radar images segmentation by region growth and texture analysis. This method is used in [71] for the mammographic images preprocessing especially for the suppression of pectoral muscle.

II.2.2.2. Regions' Divisions

This approach introduced by Samet [72] consists to divide the image into homogeneous regions according to a given criterion. We start with the whole image to be segmented; we divide image into four sub-regions if it does not satisfy the homogeneity criterion. Each sub-region is analyzed in the same way in a recursive manner until stop criterion is satisfied.

Chang [73] used the regions' division followed by extraction of textural characteristics for images' segmentation.

II.2.2.3. Regions' Fusions

Starting from each pixel of the image, which constitutes an elementary region, we gradually fusion the related regions that satisfied to appropriate criteria.

These methods use the graphs theory, in which a pre-segmented image is analyzed by a graph so-called of region adjacency (GAR) that constitutes an undirected graph in which each node represents a region and each edge defines the adjacency between two regions.

Kermad et al. [74] have proposed a textured images segmentation method which combines the multi-thresholding process followed by regions' fusions.

II.2.2.4. Decomposition / Fusion

These methods are divided into two stages: the first is the decomposition of the image into several regions and then the fusion step which consists to gradually fusion the connex regions that satisfy appropriate criteria.

The decomposition/fusion algorithm first divides image into equal size regions according to the principle illustrated in section (II.2.2.2). Once we have homogeneous regions, we fusion zones corresponding to the same region according to the principle described in section (II.2.2.3).

II.2.3. Segmentation based on pixels classification

Image classification consists in assigning to each point of the image a class chosen from a set of possible classes, whether known at the beginning or not.

The classification problem arises in different domain of the literature. In this section we present the segmentation methods based on classification that study the relationship of each pixel individually with information calculated over the entire image, we adopt a global categorization of classification algorithms, namely the unsupervised classification algorithms and supervised classification algorithms.

II.2.3.1. Supervised classification methods

A classification is said to be supervised if it requires a learning phase. This latter consists in calculating from a set of attributes' images which characterize the interest classes by involving a reference segmentation, afterwards a label is assigned to each pixel of the image from among a set defined previously (sets of classes), which means that classification is always preceded by learning. The size of the learning base has an impact on the classification rate as well as the calculation time, which means that the choice must be made with caution.

II.2.3.1.1. K-Nearest Neighbor (KNN)

This classification method was first introduced by [21]. It is a neighborhood-based method, non-parametric, which means that there is no hypothesis about classes' distributions. The principle is as follows:

Let an unknown pixel x, if the majority class in the k nearest neighbors of x is the class C_i, then the pixel is assigned to class C_i, based on the calculation of a distance (generally Euclidean distance) between the pixel candidate represented by its characteristic vector and all pixels of models textures. Results are strongly depending on the chosen value of k, the larger the k value, the longer the calculation time and the smaller the classification error. We will then choose the k value which minimizes the classification error. The KNN algorithm steps are described in algorithm II.1.

One of the major drawbacks of this classifier type is the computation time required to find the k nearest neighbors throughout the learning base and then the calculation of the distance between each new observation with the prototypes. However, it remains a reference method given its simplicity and its effectiveness in several applications fields [48,75].

Algorithm II.1: *KNN*

Inputs:

- Individual to classify x.
- L sample of learning consisting of observations x_i relating to a class C_i
- Number of neighbors k.

Output: The membership class C_x of individual x.

Begin

1- Calculation of the distance d between the individual x and each individual in learning base.
2- Search for k nearest individuals matching which minimize d.
3- Choice of the class C_x according to the majority vote or weighted vote.

End

II.2.3.1.2. K-means

K-means defined by McQueen [76] is one of the simplest algorithms for automatic data classification. The main idea is to randomly choose a set of centers fixed a priori, and iteratively search for the optimal partition. Each individual is assigned to the nearest center, after the allocation of all data the average of each group is calculated, it constitutes the new representatives of groups, when they have reached a stationary state (no data changes group) the algorithm is stopped.

Algorithm II.2: *K-means*

Inputs: x: set of N data.
k: number of desired groups.

Output: Partition of k groups $\{C_1, C_2, \dots C_k\}$.

Begin

1- Random initialization of C_k centers.
Repeat
2- Assignment: Generate a new partition by assigning each object to the group with nearest center.

$$x_i \in C_k \ \ if \ \ \forall j \ |x_i - \mu_k| = min|x_i - \mu_j| \qquad II.1$$

with μ_k is the center of class k

3- Representation: Calculate the centers associated to the new partition.

$$\mu_k = \frac{1}{N} \sum_{x_i \in C_k} x_i \qquad II.2$$

Until convergence of the algorithm to a stable partition.

End

This process tries to maximize the intra-class similarity represented as an objective function; in the case of Euclidean distance this function is called quadratic error function:

$$J = \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - C_i\|^2 \qquad II.3$$

II.2.3.1.3. Neural networks

The human brain is able to adapt, learn and decide, and this is why researchers have been interested in understanding its operating principle and being able to apply it to computer science field. Thus, in the fifties they formalized the neuron into a mathematical model from the biological model [77].

a- Biological model

The biological neuron is a living cell specialized in the processing of electrical signals. Neurons are linked together by connections called axons. These axons will themselves play an important role in the logical behavior of the whole neurons. These axons conduct electrical signals from the output of one neuron to the input (synapse) of another neuron. Neurons add up signals received at the input and, depending on the obtained result, provide a current at the output [77].

Figure II.1 shows the scheme of a biological neuron:

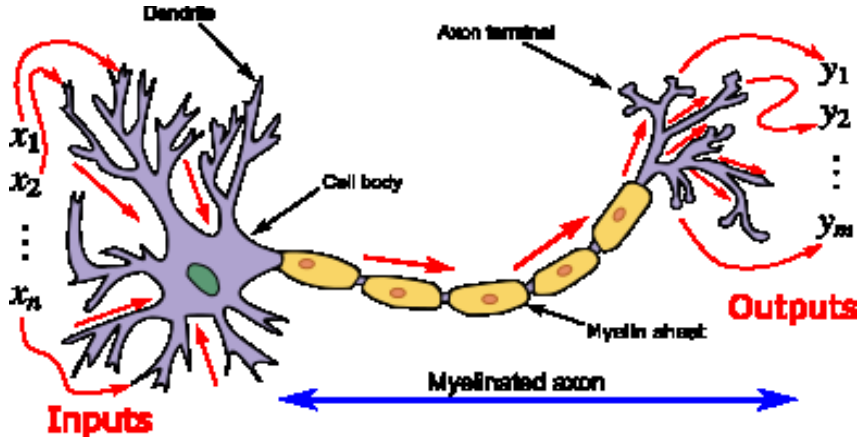


Figure II.1: Structure of biological neuron.

We can decompose it into three main regions:

The cell body: It contains the neuron core as well as the biochemical machine necessary for the synthesis of enzymes. This spherical or pyramidal cell body also contains the other molecules essential to the life of the cell. Its size is a few microns of diameter.

Dendrites: These are thin tubular extensions which branch out around the neuron and form a sort of large tree structure. Signals sent to the neuron are captured by dendrites. Their size is a few tens of microns in length.

Axon: It is along the axon that signals leave the neuron. Contrary to dendrites that branch around the neuron, the axon is longer and branches at its end where it connects to dendrites of the other neurons. Its size can vary between a few millimeters to several meters.

Synapse: A synapse is a junction between two neurons, and generally between the axon of one neuron and a dendrite of another neuron (but there are also axonal synapses for example).

b- Artificial model

What is usually referred to as a "neural network", is an artificial neural network based on a simplified model of the neuron. This model allows certain brains' functions, such as associative memorization, learning by example, working in parallel, etc. However, the formal neuron does not have all capacities of biological neurons, such as the sharing of synapses, membrane activation or the prenatal structuring of neurons, therefore the current neural networks are far from having the brain possibilities [77]. Figure II.2 shows the duality between artificial neuron and biological neuron.

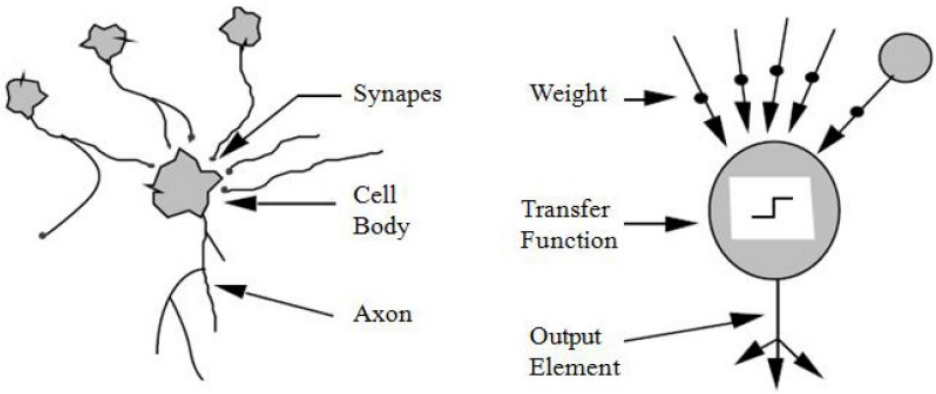


Figure II.2 Analogy between biological and artificial neuron

c- Formal neuron: The first formal neuron appeared in 1943 (Mac Culloch and Pitts). The formal neuron is therefore a mathematical modeling which takes up the principles of the biological neuron functioning, in particular the summation of inputs. Knowing that at biological level, synapses do not all have the same "value" (connections between neurons being more or less strong), the authors have therefore created an algorithm which weights the sum of its inputs by synaptic weights (weighting coefficients).

d- Mathematical interpretation: From a mathematical point of view, the formal neuron can be represented by the following way:

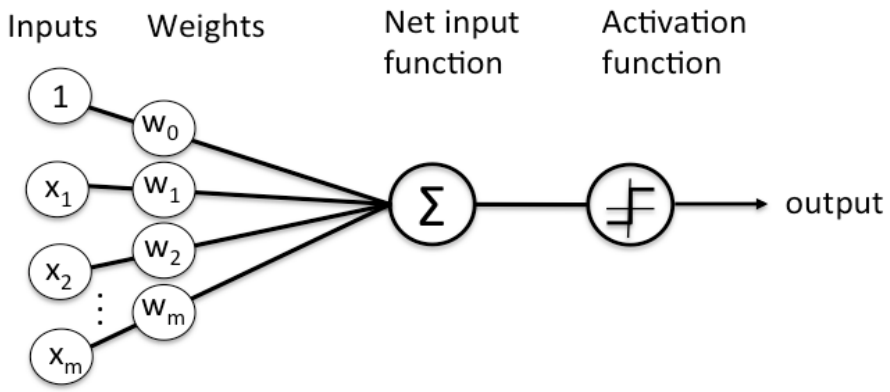


Figure II.3: Mathematical representation of the formal neuron.

For a number between j $(= 1)$ and any number m, the formal neuron will calculate the sum of its inputs $(x_1, \ldots, x_m)$, weighted by the synaptic weights $(w_1, \ldots, w_m)$, and compare it to its threshold θ. If the result is greater than the threshold, then the returned value is 1, otherwise the returned value is 0. Hence the formula:

$$y = f\left(\sum_{j=1}^{m}(w_j x_j - \theta)\right) \qquad II.4$$

Where f is the threshold function.

e- Activation functions:

Each neuron calculates its output value from the weighted sum of its inputs and its weights, there are different activation functions to calculate this value. In its first version, the formal neuron was therefore implemented with a threshold function, but many versions exist. Thus, McCulloch and Pitts' neuron has been generalized in different ways, by choosing other activation functions, such as linear or sigmoid functions for example.

Table II.1 summarizes the different types of the most used activation functions, with their mathematical equations [78].

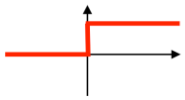
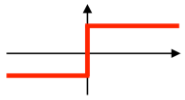
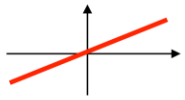
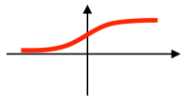
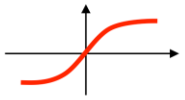
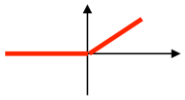
Activation function	Equation	Example	1D Graph
Unit step (Heaviside)	$\phi(z) = \begin{cases} 0, & z < 0, \\ 0.5, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Sign (Signum)	$\phi(z) = \begin{cases} -1, & z < 0, \\ 0, & z = 0, \\ 1, & z > 0, \end{cases}$	Perceptron variant	
Linear	$\phi(z) = z$	Adaline, linear regression	
Piece-wise linear	$\phi(z) = \begin{cases} 1, & z \geq \frac{1}{2}, \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2}, \\ 0, & z \leq -\frac{1}{2}, \end{cases}$	Support vector machine	
Logistic (sigmoid)	$\phi(z) = \frac{1}{1 + e^{-z}}$	Logistic regression, Multi-layer NN	
Hyperbolic tangent	$\phi(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}}$	Multi-layer Neural Networks	
Rectifier, ReLU (Rectified Linear Unit)	$\phi(z) = max(0, z)$	Multi-layer Neural Networks	

Table II.1: Summary table of different activation functions types.

f- Learning:

Learning is likely the most interesting property of neural networks. It is a phase of neural networks development during which the behavior of the network is modified until the desired behavior is obtained. Neural learning uses examples of behavior [79].

In the majority of current algorithms, the modified variables during the learning phase are the connection weights. Learning is the modification of network weights in order to match the network's response to examples and experience. At the learning end, weights are fixed: this is then the use phase.

g- Learning procedure:

Learning a network is generally done in the context of a task or behavior to be learned. Information to be processed is coded in the form of a vector called input pattern, which is communicated to input neurons of the network. Network response is interpreted from the activation value of its output neurons, whose vector is called output pattern. During supervised

learning, we also have the reference behavior that the network must learn, expressed in the form of reference pattern, or desired outputs pattern.

In general, learning takes place over a relatively long period, during which input patterns (and eventually desired patterns) can be presented to the network a large number of times. Steps for the neuron network learning are described in figure II.4.

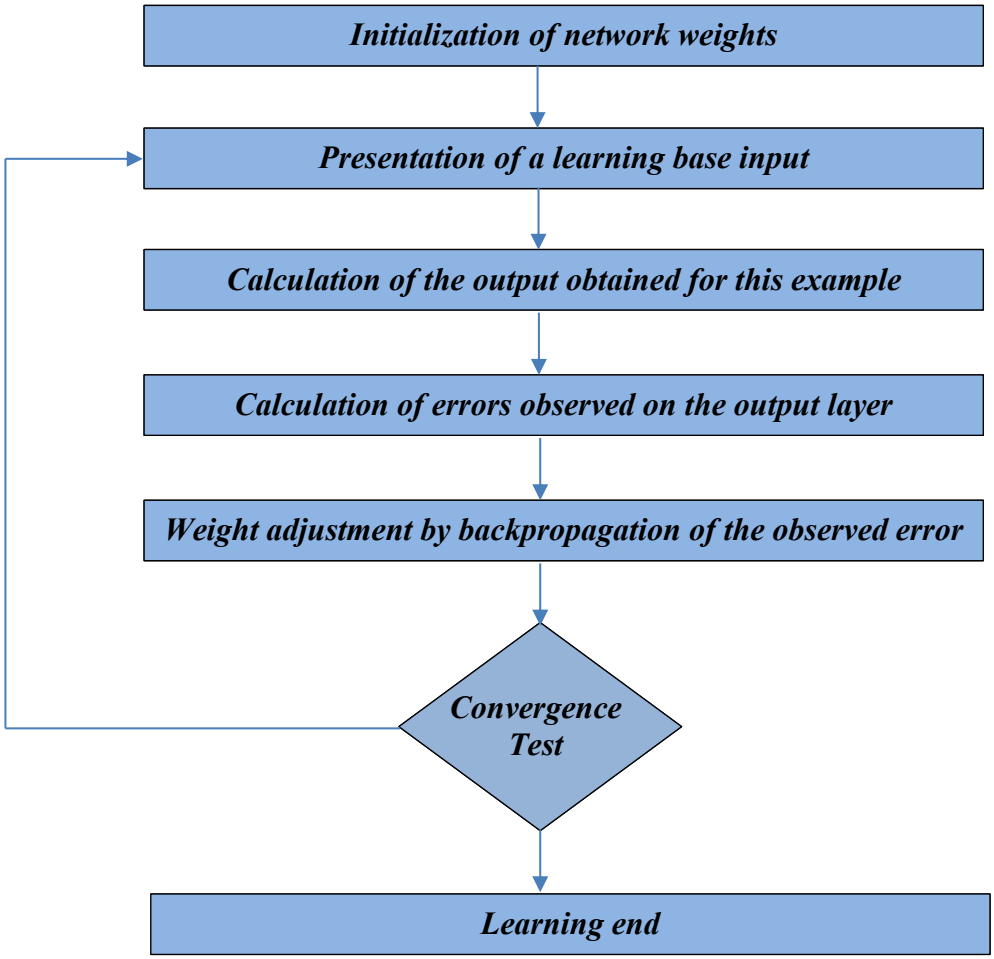


Figure II.4: Learning procedure flowchart.

h- Multilayer Perceptron (MLP): It is a network with one or more hidden layers between input and output. Each neuron in a layer is connected to all neurons in previous and next layer and there are no connections between neurons in the same layer. The activation functions used in this type of network are mainly threshold or sigmoid functions. It can solve non-linearly separable problems and more complicated logical problems, and in particular the famous problem of XOR. He also follows supervised learning according to the error correction rule (See appendix A).

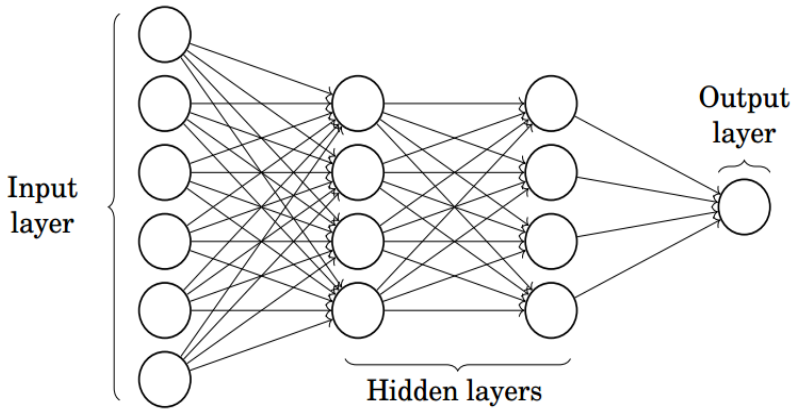


Figure II.5: Multilayer perceptron (with three layers).

i- Learning by gradient's backpropagation

Multilayer networks use the gradient's backpropagation rule (described in appendix A). Transfer functions must therefore be differentiable, this is why we use sigmoid functions, which are an infinitely differentiable approximation of the Heaviside threshold function. The principle is simple, it calculates the MLP's output according to a series of samples, and calculate the error by comparing MLP's output with the expected one, the error is then corrected [77].

II.2.3.2. Unsupervised classification methods

Classification is said to be unsupervised if it does not require any prior knowledge of the membership classes. Therefore, it does not require a learning phase. In this thesis, we will be interested in supervised classification.

II.3. Classification assessment

In order to evaluate classification systems and compare their robustness for a given application, a classic approach consists in using the confusion matrix [53].

A confusion matrix is a technique for summarizing the performance of a classification algorithm. Calculating a confusion matrix can give you a better idea of what your classification model is getting right and what types of errors it is making [80].

II.3.1. Classification Accuracy and its Limitations:

Classification accuracy is the ratio of correct predictions to total predictions made.

$$\text{Classification accuracy} = \frac{\text{Correct predictions}}{\text{Total predictions}} \times 100 \qquad II.5$$

Classification accuracy can also easily be turned into a misclassification rate or error rate by inverting the value, such as:

$$\text{Error rate} = \left(1 - \left(\frac{\text{Correct predictions}}{\text{Total predictions}}\right)\right) \times 100 \qquad II.6$$

The main problem with classification accuracy is that it hides the detail you need to better understand the performance of your classification model. For instance, with 3 or more classes you may get a classification accuracy of 80%, but you don't know if that is because all classes are being predicted equally well or whether one or two classes are being neglected by the model. But thankfully we can avoid this last problem by using a confusion matrix [80].

The confusion matrix shows the way in which our classification model is confused when it makes predictions. It gives us insight not only into the errors being made by our classifier but more importantly the types of errors that are being made. It is this breakdown that overcomes the limitation of using classification accuracy alone. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key of the confusion matrix [80].

It is a two-dimensional matrix, indexed in one dimension by the true class of an object and in the other by the class that the classifier assigns [81]. When outputs are two classes, confusion matrix is a table with 4 different combinations of predicted and actual values (as shown in Table II.2):

	Class 1 Predicted	***Class 2 Predicted***
Class 1 Actual	***TP***	***FN***
Class 2 Actual	***FP***	***TN***

Table II.2. Confusion matrix when outputs are two classes.

With:
- Class 1: Positive
- Class 2: Negative
- Positive (*P*): Observation is positive (for example: is an apple).
- Negative (*N*): Observation is not positive (for example: is not an apple).
- True Positive (*TP*): Observation is positive, and is predicted to be positive.

• False Negative (*FN*): Observation is positive, but is predicted negative.
• True Negative (*TN*): Observation is negative, and is predicted to be negative.
• False Positive (*FP*): Observation is negative, but is predicted positive.

II.3.2. Classification Rate/Accuracy: Classification Rate or Accuracy is given by the relation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad II.7$$

However, there are problems with accuracy. It assumes equal costs for both kinds of errors. A 99% accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem.

II.3.3. Recall: Recall can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (small number of *FN*). Recall is given by the relation:

$$Recall = \frac{TP}{TP + FN} \qquad II.8$$

II.3.4. Precision: To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labeled as positive is indeed positive (small number of *FP*). Precision is given by the relation:

$$Precision = \frac{TP}{TP + FP} \qquad II.9$$

High recall, low precision: This means that most of the positive examples are correctly recognized (low *FN*) but there are a lot of false positives.

Low recall, high precision: This shows that we miss a lot of positive examples (high *FN*) but those we predict as positive are indeed positive (low *FP*).

II.3.5. F-measure: Since we have two measures (Precision and Recall), it is helpful to have a measurement that represents both of them. We calculate an F-measure which uses Harmonic Mean in place of Arithmetic Mean as it punishes more the extreme values. The F-Measure will always be nearer to the smaller value of Precision or Recall.

$$F - measure = \frac{2 \times Recall \times Precision}{Recall + Precision} \qquad II.10$$

II.3.6. The kappa index of agreement:

The kappa index of agreement for categorical data was developed by Cohen [82,83] and associates in the context of psychology and psychiatric diagnosis [84]. The basic idea behind kappa is that some of the apparent classification accuracy could be due to chance. It is used to control only those instances that may have been correctly classified by chance.

Kappa Statistic compares the accuracy of the system to the accuracy of a random system. The formula to calculate Cohen's kappa is:

$$Kappa = \frac{Total\ Accuracy - Random\ Accuracy}{1 - Random\ Accuracy} \qquad II.11$$

Total accuracy is simply the sum of true positive and true negatives, divided by the total number of items, that is:

$$Total\ Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad II.12$$

Random Accuracy is defined as the sum of the products of reference likelihood and result likelihood for each class. That is,

$$RandomAccuracy = \frac{ActualFalse \times PredictedFalse + ActualTrue \times PredictedTrue}{Total \times Total} \qquad II.13$$

In terms of false positives, true Negative … etc, random accuracy can be written as:

$$RandomAccuracy = \frac{(TN + FP) \times (TN + FN) + (FN + TP) \times (FP + TP)}{Total \times Total} \qquad II.14$$

For instance, in a classification system we have obtained an ***overall accuracy*** = 89% and ***KAPPA*** = 0.86. It means that:

→ ***89%*** of the pixels are well classified.

→ ***86%*** of this good classification is not due to chance.

Indeed, the Kappa coefficient is a real number, without dimension, between -1 and +1. Table II.3 represents the degree of agreement and value of Kappa proposed by Landis and Koch [85].

Table II.3. Degree of agreement and value of Kappa proposed by Landis and Koch.

Agreement	***Kappa***
Poor	< 0
Agreement equivalent to chance	= 0
Slight	0.1 – 0.20
Fair	0.21 – 0.40
Moderate	0.41 – 0.60
Substantial	0.61 – 0.80
Near perfect	0.81 – 0.99
Perfect	= 1

Don't oppose the Kappa index with recall and precision. These last indices are not competitive of Kappa index but perfectly complementary [86].

II.4. Conclusion

In this chapter, we have presented a state of art on the most commonly used methods of textured images segmentation. We have given an overview of the most used classifiers in the state of art, namely k-nearest neighbors, k-means, and neural networks and the adequate metrics for evaluation of the classification and final segmentation step.

In the next chapter we detail information fusion principle and the various theories used for this purpose.

CHAPTER III

Information Fusion

III.1. Introduction

Generally, a single classifier is unable to handle the large variability and scalability of the data in any problem domain. Most modern techniques of pattern classification use a mixture of classifiers and combine the decisions offered to improve the classification precision.

Information fusion was initially proposed in order to manage very large amounts of multi-source data in military field, especially in a context such as the enemy targets location, the radar images fusion, etc. However, in recent years, fusion methods have been adapted and developed for applications in signal processing.

Bloch [87] presented and defined the necessary stages of fusion in image processing, Martin [88] proposed high-level information fusion approaches, particularly for the classifiers fusion in order to classify sonar images, Chitroub [89] proposed a combining classifiers method for improving the classification of multi-source/multi-date remote sensing images, Tabassian [90] proposed a classifiers combination approach based on evidence theory in order to treat data imperfection, ... etc.

Bloch [91] defined data fusion as a combination of information from multiple sources to help decision-making. The resulting fusion image contains more information than the individual source images, because data complementarity is taken into account.

III.2. Fusion Principles

Let N images I_j come from the same scene seen through different systems. Let x be the element for which we want to assign an element C_k of the decision space $D = \{C_1, C_2, \ldots C_k\}$ (C_k determines a class in our case). The fusion is generally carried out by measurements on x extracted from I_j with respect to C_k denoted $M_j^k(x)$. All these measures are gathered in the matrix M of size $(N \times k)$, as shown in Table III.1.

	C_1	C_2	…	C_k
I_1	M_1^1	M_2^1	…	M_k^1
I_2	M_1^2	M_2^2	…	M_k^2
…	…	…	…	…
I_N	M_1^N	M_2^N	…	M_k^N

Table III.1 The matrix M.

III.2.1. Fusion process steps

The fusion process steps are defined as follows [91]:

- ***Modelling***

Consists to pass from information extracted from images to a mathematical representation linked to a particular theory, that is to say choosing the formalism of elements to be fusion by defining the M_j^k's form, which can be a distribution, a cost function, a formula, etc.

- ***Estimation***

Consists of estimating the model's parameters, this step is dependent on the modeling. It is for example the estimation of distributions.

- ***Combination***

The combination step is the information gathering phase. It is the fusion heart. It consists of choosing an operator adapted to the modelling formalism.

Combination operators provide a same nature result as the combined information. The information thus provided is interpreted in the same way as the initial information.

- ***Decision***

The final step in information fusion is the decision. The choice of criterion is made according to the choice of the modelling and the combination. Conventionally, this is the minimization or the maximization of a function resulting from the combination. This step must therefore provide the expert the "best" decision d_k.

III.2.2. Information fusion manners

Knowing that there are different ways to combine information from multiple sources, four principles are possible depending on the matrix M:

- ***Global principle***:

Takes into account the global matrix M.

- ***Horizontal principle***

Makes local decisions on each image separately, then a global decision on x (column of M). The interest of such approach is that information from sources is not necessarily available simultaneously. This makes it easy to add a source. However, this approach does not take into account the possible relationships (dependencies, correlations) between sources.

- Vertical principle

Combines all measures relating to the same class C_i, then a global decision is done on x (line of M). However, as for the global fusion, the whole information must be present simultaneously.

- Intermediate principle

Adaptively chooses the necessary information for a given problem according to the image specifics.

Among all these principles, the vertical principle is the most used, thus we will be interested in this principle later.

III.2.3. Fusion levels

B.V. Dasarathy [92] proposed three fusion levels which are data, characteristics, and the decisions.

Data fusion (or low-level fusion) corresponds to the information fusion directly at the output of the sensor or the source. Data is therefore for example a coefficient of a signal at a given frequency, or even a pixel if the signal has been reconstructed in an image form.

Characteristics fusion (or medium-level fusion) is the fusion of information extracted from data coming directly from the source. The characteristics are therefore, for example, the texture parameters of an image.

Finally, the decisions fusion (or high-level fusion) relates to the fusion of information corresponding to the formulation of hypotheses from an expert or a system (for example a classifier). These three levels have often been resumed [93].

III.3. Information imperfections

One of the important characteristics of information in fusion is its imperfection. It can take various forms, which are briefly described below. The main imperfections considered are uncertainty and imprecision [94-97]. To these two imperfections can be added incompleteness [98-104], redundancy and complementarity [91,105], conflict [100,104,106,107,108], and finally ambiguity [91].

a- Uncertainty:

Uncertainty relates to the information truth, and characterizes its conformity degree to reality [109]. It refers to the nature of object or fact concerned, its quality, its essence or its occurrence. Uncertain information therefore describes a partial knowledge of reality, while what is certain

necessarily leads to knowledge of all reality. To measure uncertainty, reality must therefore be known.

b- Imprecision:

The imprecision of information is characterized by the information content. It relates to the information or to the source. It therefore measures a quantitative fault of knowledge. Reality must therefore be known or estimated.

c- Incompleteness:

Incompleteness is the lack of information provided by the source. Incomplete information can be the cause of uncertainty and imprecision. It can be measured by the difference between information amount actually provided by the source and that the source must provide. However, the information amount is difficult to quantify. One of the first measures of information quantity was introduced by C.E. Shannon [110], known as Shannon entropy.

d- Conflict:

The conflict characterizes two or more information leading to contradictory interpretations and therefore incompatible.

Conflict situations are common in fusion issues, and always pose difficult problems to solve. First of all, conflict detection is not necessarily easy. They can easily be confused with other imperfection types, or even with the complementarity of sources. Then, their identification and typology is a question which often arises, but in a different way according to their field. Finally, their resolution can take different forms. It can be based on the elimination of untrusted sources, on taking into account additional information, etc.

In some cases, it may be better to delay the combination and wait for other information that may resolve the conflict, or even not to fusion at all.

e- Ambiguity:

The ambiguity of information is the fact that it leads to two or more interpretations. Ambiguity can come from another information imperfection (uncertainty, imprecision, conflict, ...), but not necessarily. For example, the description of a radar target (by its signature) does not always make it possible to distinguish this target from another, when signatures are too close, this is the case of an imprecise signature. The fusion of information from another source can help to remove this ambiguity.

Other characteristics of information are more positive, and are used to limit imperfections such as:

f- Redundancy:

The redundancy of information or sources is the fact of having the same information several times. The fusion relies on redundancy of sources to confirm an information. For example, the observation of the same object by different sources can make it possible to locate the object with precision and to represent it in a space of higher dimension. Ideally, redundancy is used to reduce uncertainties and inaccuracies.

g- Complementarity:

Sources or information are complementary if they are of a different nature and make it possible to specify the system response. Complementarity is used directly in the fusion process to obtain a more complete global information and to remove ambiguities.

III.4. Fusion methods

The main information fusion methods come essentially from two modeling frameworks, probabilities and fuzzy approaches. We find on the one hand possibilities theory from the fuzzy subsets' theory, the belief theory, on the other hand we find Bayesian approach from probabilities. The voting principle is one of the simplest high-level fusion methods to implement. All of these methods will be studied below.

III.4.1. Voting method

The voting method is particularly suitable for the decisions' fusion. It is based on the combination of decisions provided by classifiers on the test base classes [75,88,111].

- ***Modelling***

Each source I is associated with an indicator function:

$$M_k^j(x) = \begin{cases} 1 & if \quad I_j(x) = k \\ 0 & \quad elseif \end{cases} \qquad III.1$$

Note that $I_j(x) = k$ the fact that the source I_j decides d_k, for example assigns the class C_k to the observation x.

- ***Estimate***

α_{jk} is the reliability of a source for a given class. It can be estimated from confusion matrices of each classifier.

$$\sum_{j=1}^{N} \alpha_j = 1 \qquad III.2$$

- ***Combination***

The sources' combination is written by:

For all k,

$$M_k(x) = \sum_{j=1}^{N} \alpha_{jk} M_k^j(x) \qquad III.3$$

- ***Decision***

The final vote decision corresponds to the class k most voted by individual classifiers:

$$D = argmax_{k=1\ldots C} M_k(x) \qquad III.4$$

However, this simple rule does not always admit solutions in the decisions' set. Indeed, for example if the number of sources N is pair and $\frac{N}{2}$ sources decide C_1 and $\frac{N}{2}$ other sources say C_2, or even in the case where each source assigns to x a different class.

In order to overcome this problem, an improved version designated by majority voting has been introduced, written by:

$$D = \begin{cases} k & if \quad max_k M_k(x) > \frac{N}{2} \\ D_{N+1} & elseif \end{cases} \qquad III.5$$

Where D_{N+1} represents the total uncertainty linked to the classifiers.

The voting approach advantages are double, it is very simple and natural and it does not require prior knowledge. The basic rule can however be modified so as to integrate data imperfections in the form of reliability. This method allowed to show theoretically that the fusion of classifiers can give, under certain hypotheses, better performance than each classifier taken individually. We have also noticed that it is more interesting for this approach to consider an odd number of classifiers than an even number. In practice, it is therefore more interesting to consider only an odd number of sources to increase performance.

Many studies have compared majority voting in the case of classification or recognition [112]. Even if this rule remains simple, it often gives comparable results to more complex approaches.

III.4.2. Evidence theory

The evidence theory, also known as Dempster-Shafer theory or theory of belief functions, was proposed by Dempster in 1967 [113] and then mathematically formulated by Shafer [114]. This theory was taken up in 1990 by Smets [115] under the name of transferable belief model.

The evidence theory allows the data modelling that presents inaccuracies as well as uncertainties.

III.4.2.1. Math Basics

- Discernment framework:

A discernment framework is the all possible hypotheses set of the studied problem.

Let Ω denote this set and suppose that it is composed of N different hypotheses.

$$\Omega = \{H_1, H_2, \dots, H_N\} \qquad III.6$$

Only one hypothesis of this set is considered true. The power set of the discernment set is the collection which contains all possible combinations of hypotheses. The power set is defined as follows:

$$2^{\Omega} = \{A/A \subseteq \Omega\} = \left\{\emptyset, \{H_1\}, \{H_2\}, \dots, \{H_1, H_2\}, \{H_1, H_3\}, \dots, \{H_1, \dots, H_N\}\right\} \qquad III.7$$

"A" may be a simple hypothesis or a class of hypotheses or even an empty set.

- Mass function:

Generally, a mass function, also known as a "belief mass" or BPA (Basic Probability Assignment), is simply noted m. This mass is calculated for any element of the 2^{Ω} set and is defined as follows:

$$\forall A \in 2^{\Omega}, m(A) \in [0,1],$$

$$\sum_{A \in 2^{\Omega}} m(A) = 1 \qquad III.8$$

An element which has a non-zero mass is said a focal element.

- Mass transformations:

From the mass function m we can deduce other functions such as credibility and plausibility to better represent the knowledge:

a- Credibility

The credibility (or *belief*) denoted by bel measures the total belief that can be attributed to a given element. It is defined by:

$$\forall A \in 2^{\Omega}, \qquad bel(A) = \sum_{B \subseteq A,\ B \neq \emptyset} m(B) \qquad III.9$$

b- Plausibility

The plausibility denoted by Pl measures the maximum belief that can potentially be attributed to a given element. It is defined by:

$$\forall A \in 2^{\Omega}, \qquad Pl(A) = \sum_{A \cap B \neq \emptyset} m(B) \qquad III.10$$

- Example:

Let be a set formed by three hypotheses concerning a given problem: $\Omega = \{H_1, H_2, H_3\}$. The power set of Ω is:

$$2^{\Omega} = \{\emptyset, \{H_1\}, \{H_2\}, \{H_3\}, \{H_1, H_2\}, \{H_1, H_3\}, \{H_2, H_3\}, \{H_1, H_2, H_3\}\}$$

The credibility and plausibility of the subset $A = \{H_1, H_2\}$ are calculated as follows:

$$bel(A) = m(H_1) + m(H_2) + m(\{H_1, H_2\})$$

$$Pl(A) = m(H_1) + m(H_2) + m(\{H_1, H_2\} + m(\{H_1, H_3\} + m(\{H_2, H_3\} + m(\{H_1, H_2, H_3\})$$
$$= 1 - m(H_3)$$

The interval $[bel(A), Pl(A)]$ is called the confidence interval of A .

III.4.2.2. Modelling of masses

Once we have defined the discernment set and the power set, the next step is the modelling of masses which consists to allocate belief masses to the different subsets of 2^{Ω}.

Mass modeling is a difficult step and there are no universal modeling methods that can be applied to all problems and applications. There are studies where authors define mass models specific to the processed applications. Information that is used to model masses is often a priori information, and observations on hypotheses, see Annex B for an example of mass modeling used specifically in image processing.

III.4.2.3. Combination

We assume to have N different sources noted: $S_1, S_2, \ldots, S_N$. Masses which come from each source, attributed to an element A of the discernment set, are noted respectively: $m_1(A), m_2(A), \ldots, m_N(A)$.

- Orthogonal law:

The orthogonal law or the law of Dempster combination is the normalization of conjunctive law. Let there be two sources of information (S_1, S_2), with A an element of the power set:

$$\forall A \in 2^{\Omega}, \qquad m(A) = \frac{1}{1-k} \sum_{B_1 \cap B_2 = A} m_1(B_1) m_2(B_2) \qquad III.11$$

With $m(\emptyset) = 0$ and:

$$k = \sum_{B_1 \cap B_2 = \emptyset} m_1(B_1) m_2(B_2) \qquad III.12$$

k is a law normalization term also known as a conflict term. This law has the properties of commutativity and associativity. In the case of N different sources, it is written:

$$\forall A \in 2^{\Omega}, \qquad m(A) = \frac{1}{1-k} \sum_{B_1 \cap B_2 \cap \ldots \cap B_N = A} \left(\prod_{j=1}^{N} m_j(B_j) \right) \qquad III.13$$

With:

$$\forall A \in 2^{\Omega}, \qquad k = \sum_{B_1 \cap B_2 \cap \ldots \cap B_N = \emptyset} \left(\prod_{j=1}^{N} m_j(B_j) \right) \qquad III.14$$

See annex C for an example on the use of this rule in the case of two information sources.

III.4.2.4. Decision

The final fusion step is the decision. This is the step where only one hypothesis is chosen among the discernment set hypotheses after the fusion of knowledge brought by different sources. In Bayesian case, the maximum of posterior is the decision rule while the evidence theory offers a set of different decision rules which do not necessarily give the same result:

a- Maximum of credibility

Credibility (*bel*) is interpreted as the lower bound of probability. Decision based on maximum of credibility is therefore a pessimistic decision, which is expressed as follows:

$$Dec = \arg\max bel(H), \quad H \in \Omega \qquad III.15$$

bel is the credibility function associated to the mass of the hypothesis $H \in \Omega$ corresponding.

b- Maximum of plausibility

Plausibility (*Pl*) is interpreted as the upper bound of probability. The decision of plausibility's maximum is then an optimistic decision.

$$Dec = \arg\max Pl(H), \quad H \in \Omega \qquad III.16$$

Pl denotes the plausibility function associated to the mass of the hypothesis H.

III.4.3. Possibilities theory

The possibilities theory is induced by the fuzzy sets theory introduced by Zadeh [116]. This theory was essentially developed by Dubois and Prade in France [117]. It is based on the possibility distribution function which makes it possible to deal with and model both uncertainty and imprecision. This theory has been the subject of several scientific works [118-121].

III.4.3.1. Foundations of possibility theory

- Universe of discourse:

Let Ω be the universe of discourse which presents the set of all possible hypotheses modeling the fusion problem:

$$\Omega = \{\omega_1, \omega_2, \dots, \omega_n\} \qquad III.17$$

- Distribution of possibilities:

The distribution of possibilities noted π is the fundamental function of the possibilities theory. It associates to each hypothesis of Ω a degree of possibility.

The distribution of possibilities is a function of ω in [0,1] which verifies:

$$sup_{\omega\in\Omega}\{\pi(\omega)\} = 1 \qquad III.18$$

In other words, there exists at least one event of Ω such that its degree of possibility is equal to 1. With $\pi(w)$ presents the degree of possibility for the observation x to be equal to ω, knowing that:

- If $\pi(\omega) = 0$, this indicates that x cannot be equal to ω.
- If $\pi(\omega) = 1$, this indicates that nothing prevents x from being worth ω.

A precise information is modeled by the distribution of possibilities as indicated in the following expression:

$$\exists!\ \omega_0 \in \Omega \quad \text{such as} \begin{cases} \pi(\omega_0) = 1 \\ \pi(\omega) = 0 \quad \forall\ \omega \neq \omega_0 \end{cases} \qquad III.19$$

However, we are in the total ignorance case when:

$$\forall\ \omega \in \Omega,\ \pi(\omega) = 1 \qquad III.20$$

- Measure of possibility:

A measure of possibility noted $\prod$ is a function of 2^{Ω} (the power set of Ω) in [0,1] which verifies the following equations' system:

$$\begin{cases} \prod(\emptyset) = 0 \\ \prod(\Omega) = 1 \\ \forall\, A \in 2^{\Omega},\ \prod(A) = sup\,\{\pi(x), x \in A\} \end{cases} \qquad III.21$$

The closer the possibility value of the event is to 1, the more its realization is possible. If the possibility of an event is zero it means that this event is impossible. The measurement and distribution of possibility makes it possible to quantify and model the information inaccuracy.

- ***Measure of necessity:***

A measure of necessity denoted N is a function of 2^{Ω} in [0,1] that satisfies the following equations:

$$\begin{cases} N(\emptyset) = 0 \\ N(\Omega) = 1 \\ \forall\, A \in 2^{\Omega},\ N(A) = inf\,\{1 - \pi(x), x \notin A\} \end{cases} \qquad III.22$$

The realization's probability of an event A is delimited by the necessity measure $N(A)$ and the possibility measure $\prod(A)$, (i.e. $N(A) \leq Pr(A) \leq \prod(A)$).

III.4.3.2. Fusion by possibility theory

- ***Modelling:***

The first task of this fusion step is to determine the different hypotheses that make up the discourse universe Ω. Suppose that we have m sources, we denote by S_j the source number j with $j \in \{1, \ldots, m\}$.

We note by $\pi_x^j(\omega_i)$ the degree of possibility for decision number i to be valid for observation x according to source S_j.

- ***Estimate:***

The construction of a possibility's distribution must be done according to available knowledge which can come in different forms (sensor data, expert opinion, ...) and therefore there is no general form.

The distribution of possibilities can take many forms. We cite the best known which are triangular shape, trapezoidal shape and Gaussian shape, as seen in figure III.1.

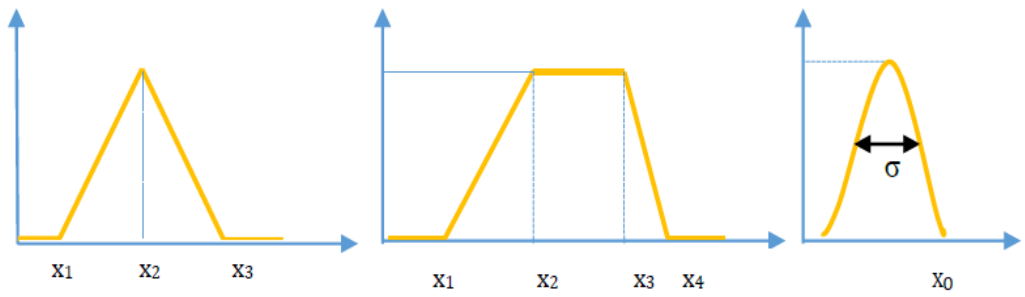


Figure III.1 Forms of possibility distributions.

In addition, several methods have been adopted to determine the distribution of possibilities such as:

1- *Histogram-possibility transformations* [122, 123].
2- *Method based on the gray level histogram* [124,125].
3- *Psychometric approaches* [126].

- Combination :

One of the major interests of possibility theory fusion is the wide variety of fusion operators allowing the combination of possibility distributions. It is common to consider conjunctive or disjunctive operators:

a- Conjunctives operators: They combine information in the way of a "logical AND" (conjunction). In this way the result will be close to 1 if and only if all values to be combined are close to 1. Among these operators the triangular norms or t-norm are the most used.

Here are some t-norms for two sources S_1 and S_2:

- Zadeh's T-norm:

$$\pi(\omega) = \min\left(\pi^{S_1}(\omega), \pi^{S_2}(\omega)\right) \qquad III.23$$

- Probabilistic T-norm:

$$\pi(\omega) = \pi^{S_1}(\omega) * \pi^{S_2}(\omega) \qquad III.24$$

- Lukasiewicz's T-norm:

$$\pi(\omega) = \max(0, \pi^{S_1}(\omega) + \pi^{S_2}(\omega) - 1) \qquad III.25$$

Conjunctives operators are generally used when sources are concordant and reliable.

b- Disjunctive operators: They combine information in the way of a "logical OR" (disjunction). The value of the combination result will be large as soon as one of the combined values is. The main disjunctive operators are the triangular conorms or t-conorms. Here are a few t-conorms applied for two sources S_1 and S_2:

- Zadeh's T-conorm:

$$\pi(\omega) = \max\left(\pi^{S_1}(\omega), \pi^{S_2}(\omega)\right) \qquad III.26$$

- Probabilistic T-conorm:

$$\pi(\omega) = \pi^{S_1}(\omega) + \pi^{S_2}(\omega) - \pi^{S_1}(\omega) * \pi^{S_2}(\omega) \qquad III.27$$

- Lukasiewicz's T-conorm:

$$\pi(\omega) = \min(\pi^{S_1}(\omega) + \pi^{S_2}(\omega) - 1,1) \qquad III.28$$

Disjunctive operators are generally used when sources are discordant and unreliable.

There are other categories of combination's operators such as: mean operators, fuzzy integrals ... etc [127, 128].

- Decision

Once information from sources are combined, the decision is made by maximizing the possibility or the necessity. Thus, we choose the decision d_k if:

$$\pi^k(x) = \max\left(\pi^i(x), i = 1 \dots n\right) \qquad III.29$$

III.4.4. Bayes Fusion Theory

Bayes fusion was one of the first techniques used to make the combination of images with decision-making. In Bayesian approaches, information's imperfections and the information itself are modeled from probability distributions or statistical measures to be estimated such as the information's amount. They were the only ones who interpreted the notion of chance and uncertainty.

Literature is very abundant for the multispectral images' segmentation using probability theory. This technique was first developed for visible [129] and aerial [130,131] imagery. Many authors have chosen this model because it has a very well-defined framework with well-known mathematical properties [132]. Manipulating probability densities is also much more familiar to the signal processing community than using somewhat newer concepts such as the Dempster-Shafer mass function notions or the membership functions of fuzzy logic. The necessary expertise in some cases is also more easily affordable via probabilities, and efficient methods are known for modeling a phenomenon by a probability density from a learning set.

III.4.4.1. Modelling:

Let $H_1,\ H_2, \dots H_N$ be a set of mutually exclusive hypotheses. They satisfy the following conditions:

$$\begin{cases} \forall\, i,j,\ H_i \cap H_j = \emptyset, & i \neq j, \\ \cup_{i=1}^{N} H_i = E, \end{cases} \qquad III.30$$

where *E* represents the hypothesis's space (that is to say the set of classes of the fused image). The hypotheses are mutually exclusive and form a partition of *E*.

III.4.4.2. Combination:

Let m_1 and m_2 be two characteristic primitives from two different images, representing the same object, or the same hypothesis H_i, the Bayesian theory calculates the likelihood of obtaining the hypothesis H_i from the two measures $m_1$and m_2 via Bayes rule [132]:

$$P(H_i/m_1, m_2) = \frac{P(H_i) \cdot P(m_1, m_2/H_i)}{\sum_{j=1}^{N} P(H_j) \cdot P(m_1, m_2/H_j)}, \qquad III.31$$

where $P(m_1, m_2/H_i)$ represents the joint probability of having both measures (m_1, m_2) when the hypothesis H_i is realized, and $P(H_i)$ is the prior probability of the hypothesis H_i, which shows the possibility of occurrences of the hypothesis H_i in the general case [132].

If m_1 and m_2 are two independent random variables, the conditional probability $P(m_1, m_2/H_i)$, also named the likelihood function, becomes a separable function of the two variables m_1 and m_2:

$$P(m_1, m_2/H_i) = P(m_1/H_i) \cdot P(m_2/H_i) \qquad III.32$$

The separability condition is appropriate in the sense that measure m_1 from one acquisition has no influence on measure m_2 from the other acquisition. Statistical independence is achieved when the two measures do not interfere with each other.

Thus equation ($III.31$) takes the following form:

$$P(H_i/m_1, m_2) = \frac{P(H_i) \cdot P(m_1/H_i) \cdot P(m_2/H_i)}{\sum_{j=1}^{N} P(H_j) \cdot P(m_1/H_j) \cdot P(m_2/H_j)} \qquad III.33$$

So, to determine the posterior probabilities $P(H_i/m_1, m_2)$, we need first to calculate the prior probabilities $P(H_i)$ for all hypotheses H_i, i ranging from 1 to N, and the likelihood functions $P(m_j/H_i)$ for every image primitive m_j and for every hypothesis.

To model the likelihood functions, generally we work under the Gaussian hypothesis [132]:

$$P(m_j/H_i) = \frac{1}{\sigma_i\sqrt{2\pi}} \exp\left(-\frac{(m_j - \overline{H_i})^2}{2\sigma_i^2}\right), \qquad III.34$$

where $\overline{H_i}$ represents the mean and σ_i is the standard deviation of the Gaussian expression.

III.4.4.3. Decision:

Once the combination of the probabilities realized by equation ($III.33$), we must choose a decision's criterion in order to decide which hypothesis H_i should be selected according to all

posterior probabilities. Several criteria are proposed in the literature: the most commonly used criterion is the maximum of posterior probability, which chooses the hypothesis H_i having the highest probability $P(H_i/m_1, m_2)$ [132].

Other criteria have also been used, but less commonly, for example:

a- The maximum expectation *(E)*, we will choose H_k if:

$$E[H_k \backslash m_1, \dots, m_l] = max_{i\in\{1,\dots,N\}} E[H_i \backslash m_1, \dots, m_l] \qquad III.35$$

b- The maximum of likelihood *(L)*, we will choose H_k if H_k is solution of:

$$\frac{\partial L(m_1, \dots, m_l, H_i)}{\partial H_i} = 0 \qquad III.36$$

With

$$\frac{\partial^2 L(m_1, \dots, m_l, H_i)}{\partial H_i^2} < 0 \qquad III.37$$

And the likelihood function is given by:

$$L(m_1, \dots, m_l, H_i) = P(H_i \backslash m_1, \dots, m_l) \qquad III.38$$

c- The maximum of entropy *(T)*, we will choose H_k if:

$$\begin{aligned} T(H_k \backslash m_1, \dots, m_l) &= max_{i\in\{1,\dots,N\}} T[H_i \backslash m_1, \dots, m_l] \\ &= max_{i\in\{1,\dots,N\}} \{-P(H_i \backslash m_1, \dots, m_l) \times \ln P(H_i \backslash m_1, \dots, m_l)\} \end{aligned} \qquad III.39$$

d- The minimax criterion, which consists in taking the decision H_k which gives the maximum performance in the most unfavorable case.

Once we have obtained a first segmentation, we can then update the parameters of different classes from this classification. Parameters of each class are therefore updated after each images segmentation. The process is repeated until convergence of the algorithm, that is to say until no pixel changes class from one iteration to another.

See annex D for an example of fusing two simulation images using a Bayesian approach.

III.4.4.4. Applications

We list in this part some works in a non-exhaustive way having used the Bayesian approach for information fusion. Let us first cite the work of L.I. Kuncheva [133,134], who compares the naive Bayesian approach to the maximum, minimum, average and the voting method. These approaches are compared in the context of classifiers' fusion. The obtained conclusion on

different databases shows the good performance of the naive approach, the average approach gives reasonable rates and keeps good stability unlike the other approaches.

Chauvin [135] studies the performance of the Bayesian fusion model for the segmentation of satellite images (in 3 classes) with the product as a combination's operator and the maximum of posterior as decision rule. He has shown that performances are a function of learning (choice of conditional probability density and a priori probabilities) and of images quality, measured by a criterion based on Fisher's intra-class inertia.

Xu et al. [112] compare the Bayesian classifier to the vote approach and to Dempster-Shafer theory for an application in handwritten recognition. It is shown that if the confusion matrices are well learned, the Bayesian approach is the best. However, this approach is unstable; performance is quickly degraded if learning is poor. The Bayesian approach is also used in images fusion [136,137].

III.5. Conclusion

In this chapter, we have presented the general notions of fusion and the different theories used for this purpose, especially the theory of uncertainty. In the next chapter we present all contributions made in this thesis, and we evaluate the performances of the proposed systems.

CHAPTER IV

Results and Discussions

IV.1. Introduction

We present in this chapter the obtained results after applying texture extraction methods followed by classification and fusion methods, presented in previous chapters, on synthetic and real images.

IV.2. Application on synthetic images

We attempt to use three types of feature extraction techniques with the purpose of classifying textured images. These features are statistical features taken out of GLCM matrices and structural features obtained using Gabor filters and the wavelet transform coefficients.

Concerning the wavelet transform, as mentioned in section I.3.3.2, the values of transformed coefficients in detail and approximation images, sub-band images, (see figure I.10) represent the necessary features that capture useful discrimination information for texture segmentation [59].

- *Choice of the wavelet:* In our work, we have used a second order biorthogonal spline wavelet. This wavelet is often used for texture analysis due to its excellent location in the frequency and spatial domains and its sensitivity to local singularity and correlation of the image [18].

As mentioned also in section I.3.3.2, the evaluation of indices is done on a sliding window *W*. The energy and the local mean on the sliding window are calculated from the resultant sub-band images. In our work, we have used a wavelet on one level, so the feature vector of each window is made of eight parameters V= [E_{LL}, E_{LH}, E_{HL}, E_{HH}, M_{LL}, M_{LH}, M_{HL}, M_{HH}], where E and M represent the energy and the local mean respectively, as seen in figure IV.1.

Various tests were carried out on a series of window sizes going from 5×5 to 25×25. The uppermost good classification rate was attained for a window of 11×11 dimension.

The estimated features vector of every window is used as an input to an appropriate classification algorithm. The MLP neural networks classifier is chosen among the most well-known classifiers, it was initiated in [37,38]. The estimated feature vector of each pixel is sent to the neural networks classifier for pre-labeling, and the score for the window provided by the neural networks is allocated to its central pixel.

For the training phase of the neural networks, images of 320×320 size were used, and 6400 patterns are selected randomly from each image, which represents 6.25% of the total number of patterns.

For the choice of the hidden layers' number and the number of neurons in each layer, we choose the rule proposed by Wierenga and Kluytmans [138] since there is no general rule other than rules of thumb as proposed in [139,140]. The size of the hidden layer is 75% of the input layer.

For the transfer functions, we retain the most used in the literature, namely the logistic function (equation $IV.1$) and the hyperbolic tangent function (equation $IV.2$).

$$\rho(n) = \frac{1}{1+e^{-n}} \qquad IV.1$$

$$tanh(n) = \frac{2}{1+e^{-2n}} - 1 \qquad IV.2$$

The gradient backpropagation algorithm is used for the training of neural networks (described in appendix A).

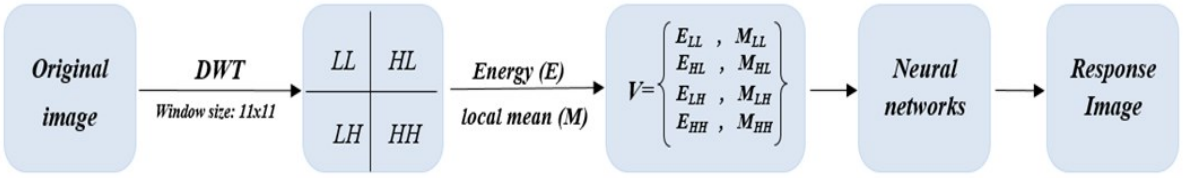


Figure IV.1 *Wavelet features extraction.*

Concerning the GLCM, a series of tests using different combinations were performed in order to fix its parameters (see section I.3.1.2.1). For both the window size and the distance d, the highest good classification rate was obtained for a 15×15 window and a distance 'd' of six-pixels. The gray levels number of images after re-quantization is set to 16 levels. These parameters will be retained for the following steps.

As mentioned also in the section I.3.1.2.1, after obtaining GLCM matrices over the sliding window in the input image, several statistics like contrast, uniformity, and entropy are extracted from these matrices [43]. Haralick suggests the calculation of fourteen features [45], not all relevant for a given application. We take here four directions: 0^0, 45^0, 90^0 and 135^0. In these four directions, six parameters are calculated and then averaged over the four directions. So, the vector of features related to each window consists of six factors V= [Homogeneity, Contrast, Entropy, Correlation, Directivity, Uniformity], (see figure IV.2). For each window, the estimated vector of features is sent to the neural networks classifier for a primary labeling, and the score for the window provided by the neural networks is assigned to its mid pixel. For the neural networks, we have used the same parameters that are used in wavelet transform.

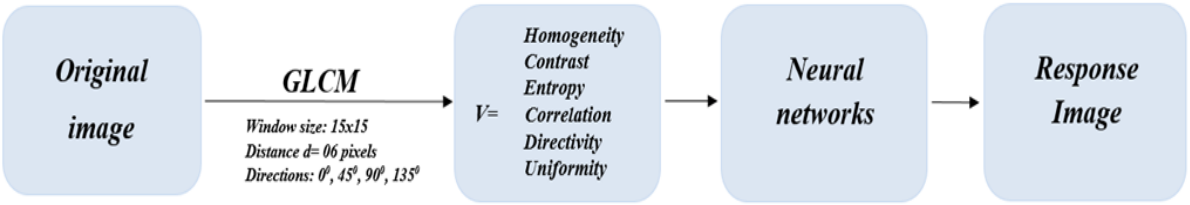


Figure IV.2 *GLCM features extraction process.*

Concerning Gabor Filters, as mentioned in section I.3.3.1, an ensemble of wavelets is created using a mother wavelet like in equation ($I.18$). Then the entire image is used as the input to the wavelet set, as shown in figure IV.3. By doing so, the input image is transformed into a set of N filtered images. As the filters output are complex valued, we exploit the magnitude of these values.

Two important parameters within Gabor filters are the number of frequencies (N_f) and directions (N_d) [43]. For the images having size of 320 x 320 used in this work, five radial frequencies ($N_f = \sqrt{2}/2^6$, $\sqrt{2}/2^5$, $\sqrt{2}/2^4$, $\sqrt{2}/2^3$, and $\sqrt{2}/2^2$) with 4 orientations ($N_d = 0^0$, 45^0, 90^0, 135^0) were adopted according to [56]. So, the features vector is composed of twenty parameters (5 frequencies x 4 orientations). Finally, the estimated features vector is sent to the neural networks classifier for an initial labeling (as seen in figure IV.3). Also, for the neural networks, we have used the same parameters that are used in wavelet transform and GLCM.

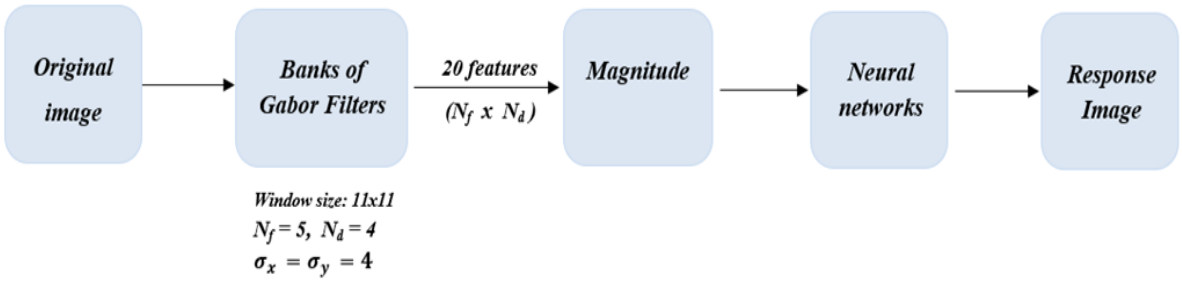


Figure IV.3 *Gabor features extraction process.*

IV.2.1 The first fusion algorithm

Combining three types of features increases the classification performance since it uses Gabor features which are more efficient in near-class borders areas combined with GLCM features which are powerful in the areas within classes, and the wavelet coefficients that offer great discriminatory power for textures with strong resemblances.

So, our first fusion algorithm uses GLCM, Gabor filters, and Wavelet transform as features extraction strategies on textured images in order to get more information in this data set. The parameters for each feature set were selected as mentioned in the previous paragraphs. After a

proper texture features extraction, each estimated feature vector of every pixel is sent to the neural networks classifier for primary labeling. Next, for each pixel a Bayes fusion method is used to combine the three scores obtained by the neural networks for each kind of features namely GLCM, Gabor and Wavelets (see figure IV.4).

Concerning Bayes fusion, as mentioned in section III.4.4.2, to determine the posterior probabilities $P(H_i/m_1, m_2)$, we need first to calculate the prior probabilities $P(H_i)$ for all hypotheses H_i, *i* ranging from 1 to *N*, and the likelihood functions $P(m_j/H_i)$ for every image primitive m_j and for every hypothesis.

So, in absence of any information on classes distribution, we assume the hypothesis of the equiprobability of different classes. If we call *K* the number of classes supposed to be present in the image, we have that:

$$\forall i,\ P(H_i) = \frac{1}{K} \qquad IV.3$$

Concerning the likelihood functions $P(m_j/H_i)$ which measure the probability of a considered gray level m_j given the hypothesis H_i, we place ourselves under the Gaussian hypothesis and we can use the following formula:

$$P(m_j/H_i) = \frac{1}{\sigma_i\sqrt{2\pi}} \exp\left(-\frac{(m_j - \overline{H_i})^2}{2\sigma_i^2}\right), \qquad IV.4$$

where $\overline{H_i}$ represents the mean and σ_i is the standard deviation of the Gaussian expression.

Once the combination of the probabilities has been realized, as mentioned in equation III.33, we use the maximum of posterior probability as decision's criterion, in order to decide which hypothesis H_i should be selected according to all posterior probabilities, which chooses the hypothesis H_i having the highest probability $P(H_i/m_1, m_2)$.

After having obtained a first segmentation, we can then update the parameters of different classes ($\overline{H_i}$, σ_i and $P(H_i)$) from this classification. The parameters of each class are re-updated after each segmentation of the images. The process is reiterated until the convergence of the algorithm, that is to say until no pixel changes class from one iteration to another.

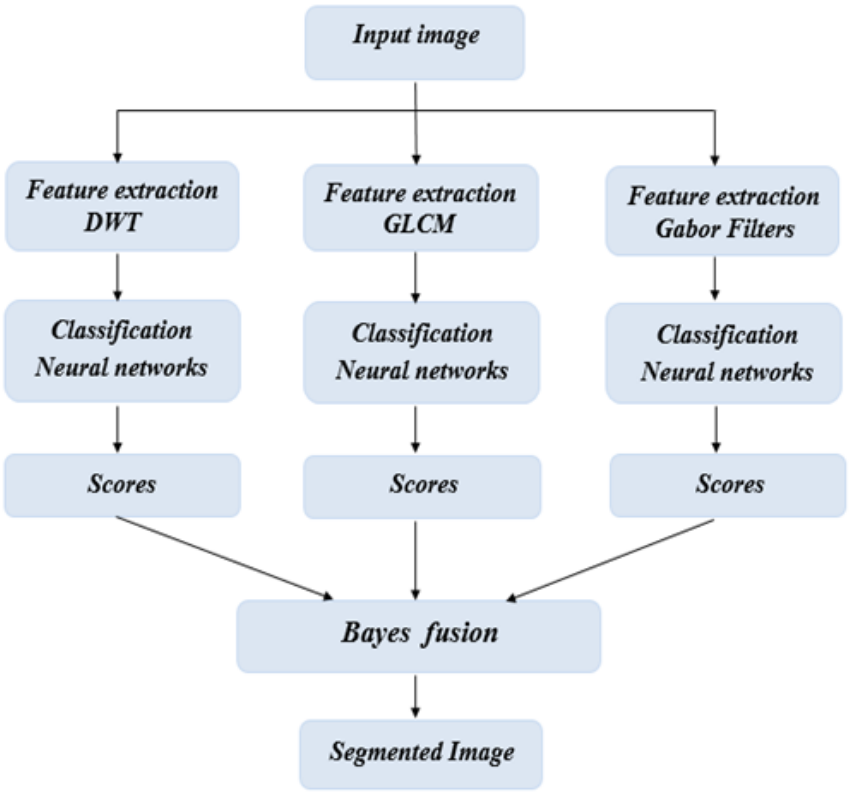


Figure IV.4 *Architecture of the proposed algorithm.*

The principle steps of the first fusion algorithm are:

Fist fusion algorithm	
Step1: Extraction of features by DWT, GLCM and Gabor filters.	
Step2: Neuronal classification of each estimated features vector. We obtain:	

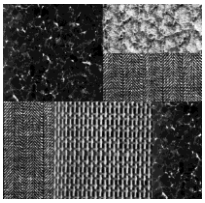
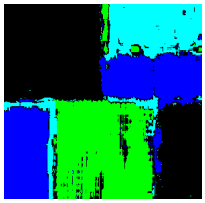
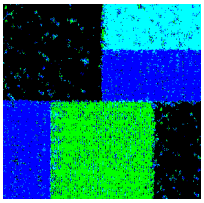
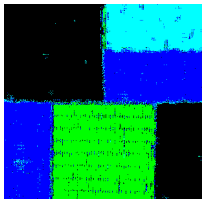
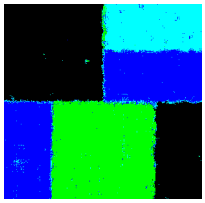
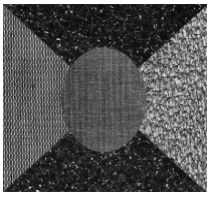
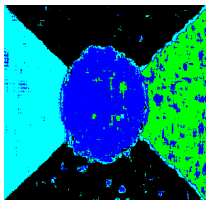
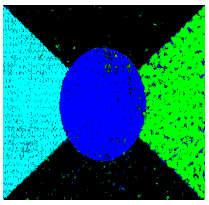
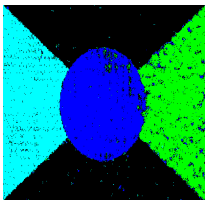
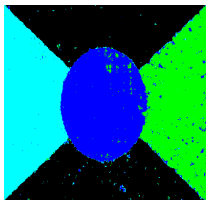
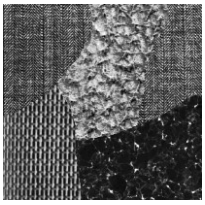
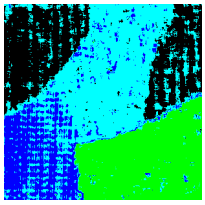
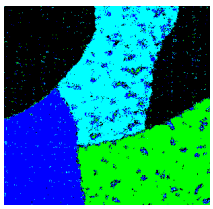
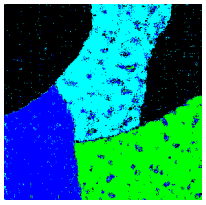
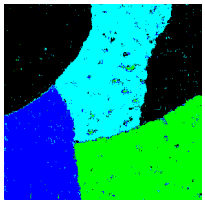
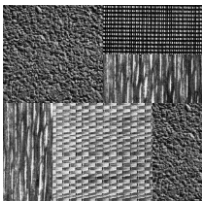
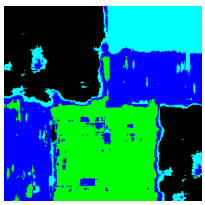
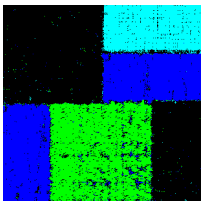
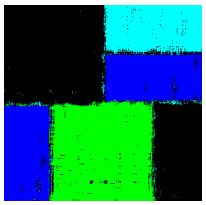
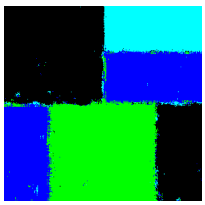
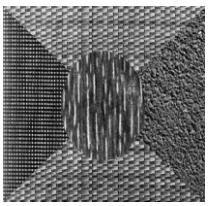
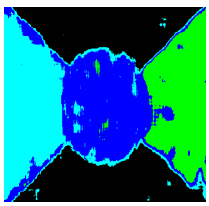
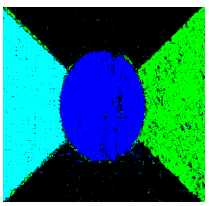
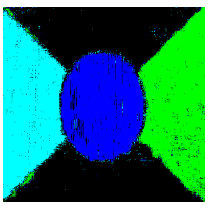
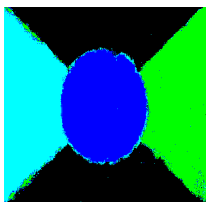
Step3:	Scores1: for DWT.
	Scores2: for GLCM.
	Scores3: for Gabor filters.
	while the recognition rate is changed **do**
	for each pixel **do**
	Bayesian fusion of Score1, 2 and 3
	end
	Calculate the recognition rate.
	end

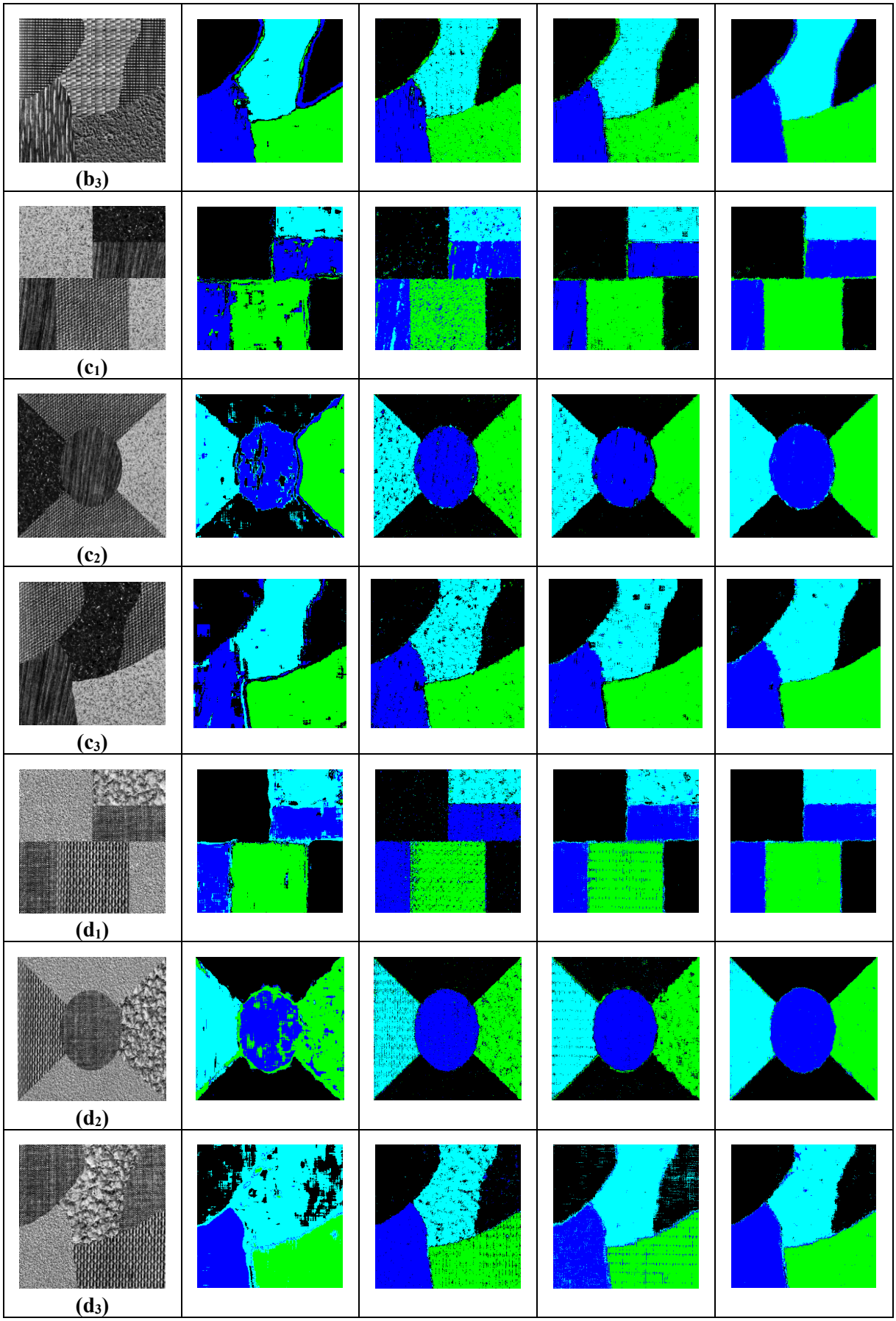
The performance of the proposed algorithms for segmenting textured images is assessed using many images with different textures and different shapes. They are collected in the wild and in cluttered conditions, from Brodatz and DTD datasets [141,142]. These last bases are constituted of stochastic (micro-textures) and regular textures (macro-textures). We have formed textured images composed of different textures taken randomly.

We have used MATLAB R2016 environment, in a Workstation (Intel(R) Xeon(R) CPU E5-1620 v3 @ 3.50 GHz 3.50 GHz, RAM: 16.00 Go) to implement our algorithms.

As shown in the first column of table IV.1. Each image (a), (b), (c), (d), contain the same textures that are present in different forms and shapes (square, rectangle, triangle, circle, and curved shape) to check the performance that could be achieved by the proposed fusion techniques.

Table IV.1 Experimental results. The first column contains textured images in different shapes. The second–fifth columns are the classification results using respectively GLCM, Gabor filters, Wavelet, 1st fusion algorithm.

Original image	*GLCM*	*Gabor filters*	*Wavelet*	*1st fusion algorithm*
 **(a_1)**				
 **(a_2)**				
 **(a_3)**				
 **(b_1)**				
 **(b_2)**				



(b3) (c1) (c2) (c3) (d1) (d2) (d3)

As shown in the 2nd, 3th and 4th columns of table IV.1 the different textures are not well differentiated when feature vectors are used individually for classification. Therefore, to improve classification precision we integrated a Bayes fusion scheme in the score obtained by the neural networks for each kind of features, the results are illustrated in the 5th column of the same Table. These last results clearly confirm the power of the fused feature vectors compared to the results obtained using individual feature vectors.

Table IV.2 demonstrates the classification accuracies that were achieved by utilizing individual, and fused feature vectors. According to the 5th column of table IV.2, the result of this first fusion algorithm shows that the quality of the segmented images is enhanced and that the classification accuracies are improved compared with the accuracies of single classifiers in all images.

Table IV.2 Classification accuracy of single classifiers compared with the integrated classifier.

Image	GLCM	Gabor filters	Wavelet	1st fusion algorithm
(a_1)	88.27 %	90.32 %	94.03 %	96.27 %
(a_2)	89.89 %	93.35 %	94.42 %	96.93 %
(a_3)	82.57 %	92.53 %	92.99 %	96.81 %
(b_1)	87.53 %	92.66 %	94.93 %	97.22 %
(b_2)	89.93 %	94.02 %	94.57 %	96.49 %
(b_3)	90.98 %	93.76 %	94.67 %	96.33 %
(c_1)	89.41 %	92.24 %	94.71 %	97.53 %
(c_2)	87.85 %	93.88 %	95.41 %	97.32 %
(c_3)	88.19 %	94.14 %	95.74 %	97.06 %
(d_1)	90.77 %	93.08 %	94.71 %	97.56 %
(d_2)	88.22 %	92.87 %	94.74 %	97.67 %
(d_3)	83.35 %	92.98 %	93.55 %	96.79 %

IV.2.2 The second fusion algorithm

In order to more improve the decision-making, a new fusion model will be examined beneath. Using a sliding window, for each kind of features, the class for this window is allocated to its central pixel. Nevertheless, this central pixel belongs to other window neighbors that may be classified into other classes. Therefore, in order to achieve a more precise segmentation result,

for each pixel a Bayes fusion method is used to combine the scores results of various windows that include this central pixel.

Let I^k be the segmented images containing the scores S_{ij}^k of each pixel (the output of the neural networks classifier) for each kind of features:

$I^k = S_{ij}^k$ with $i = 1 \ldots n$, $j = 1 \ldots m$, $k = 1, \ldots, c$,

where n and m represent the sizes of the textured image, and k is the number of different kinds of features.

We perused the images by using a sliding window of size $M \times M$, so that every pixel is surrounded by $M^2 - 1$ pixels. For each type of features, each central pixel $P_{ij,l}$ of window W_l with score $S_{ij,l}$ belongs to the $M^2 - 1$ window in the surrounding windows before the classification process. However, each central pixel $P_{ij,z}$ of the window z, with $z = 1 \cdots M^2 - 1$ produced different scores $S_{ij,z}$.

For instance, in the case of pixel $P_{3,3}$ with score $S_{3,3}$ and $M = 3$, the central pixel is surrounded by eight pixels, that are the center of the eight windows which pixel $P_{3,3}$ belonged to, (see figure IV.5).

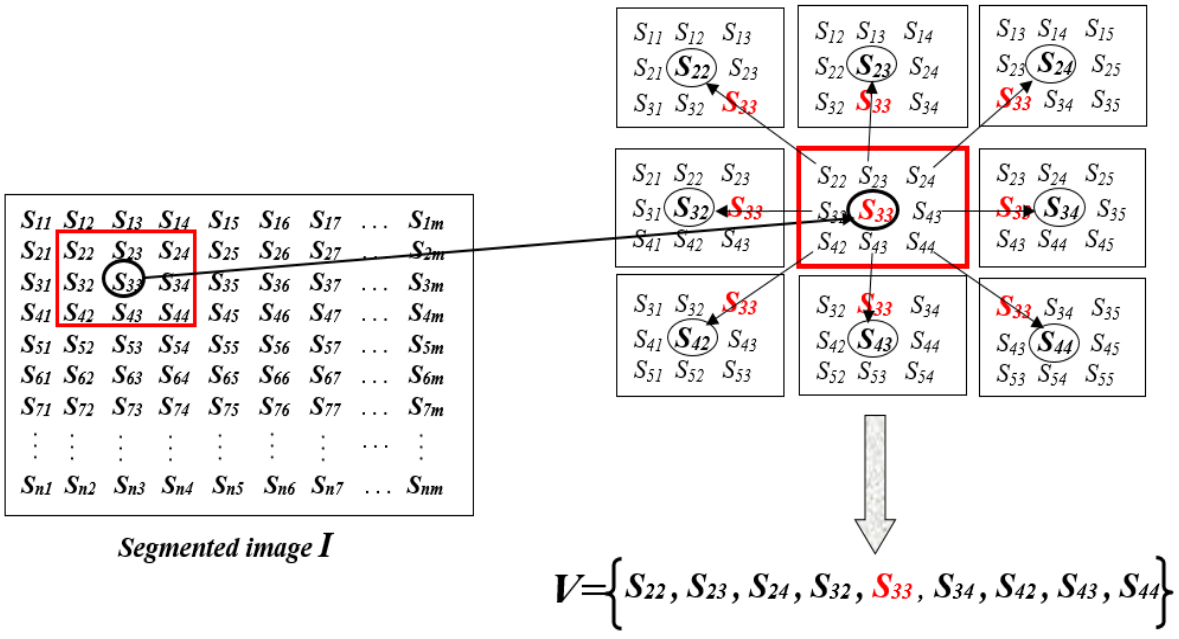


Figure IV.5 *Composition of the fusion vector.*

Moreover, it is noticed by the first fusion algorithm that the most segmentation error is situated in the borders of different classes because in these situations the mask contains more than one type of texture. So, in this new fusion model, and in order to decrease the segmentation error, we will do the fusion between only Gabor and Wavelet features due to the fact that a big mask (15×15) of GLCM features induces an increase in the classification error at the borders.

From the above example, we combined the scores produced by the current block and its eight neighboring ones for the Gabor and wavelet features: $\{S^1_{33}, S^1_{32}, S^1_{34}, S^1_{23}, S^1_{24}, S^1_{22}, S^1_{42}, S^1_{44}, S^1_{43}, S^2_{33}, S^2_{32}, S^2_{34}, S^2_{23}, S^2_{24}, S^2_{22}, S^2_{42}, S^2_{44}, S^2_{43}\}$ (see figure IV.6).

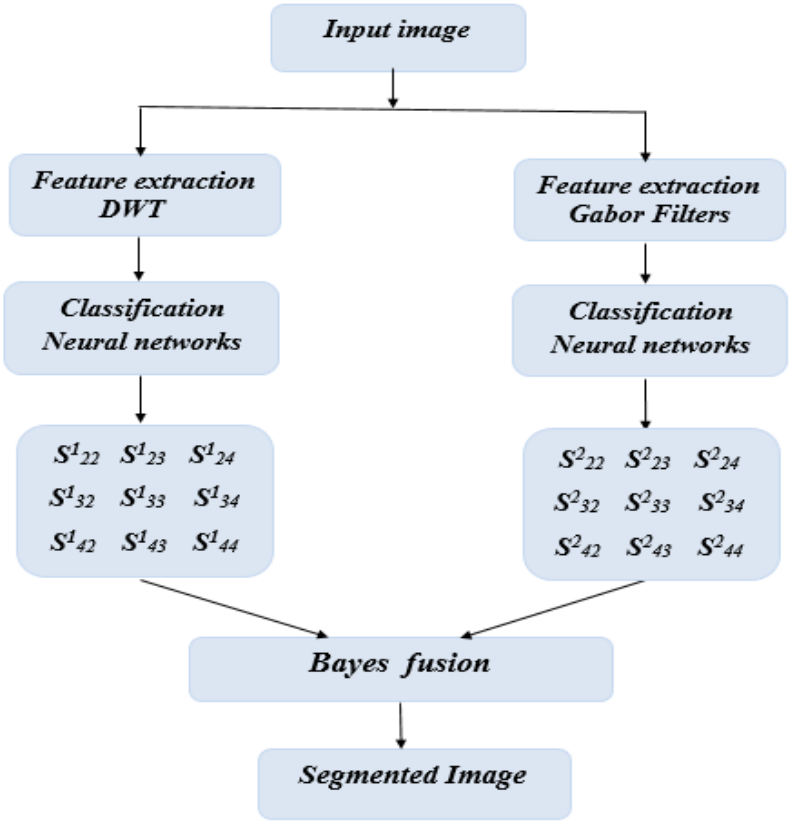


Figure IV.6 *Architecture of the second fusion algorithm.*

In this work, a sliding window of size 9×9 is used, so the central pixel is surrounded by 80 pixels, that are the center of the 80 windows which pixel $P_{9,9}$ belonged to.

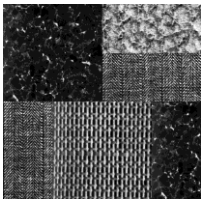
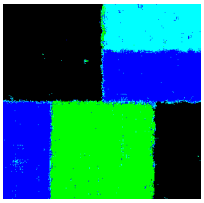
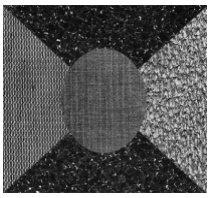
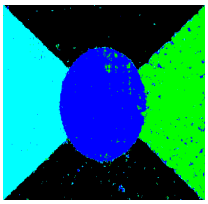
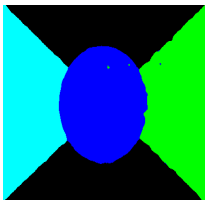
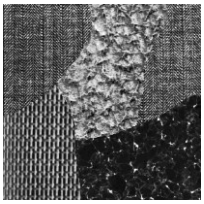
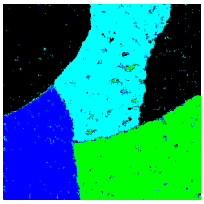
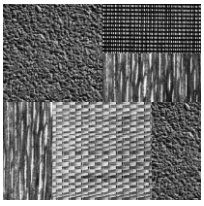
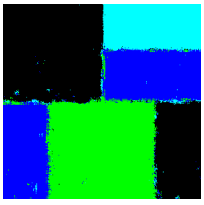
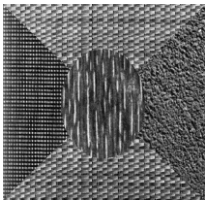
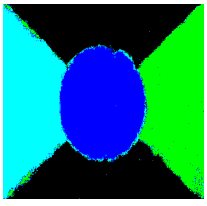
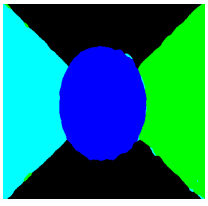
We also consider the position of the thumbnails (Windows) depending on the studied pixel, as a result, pixels belonging to the thumbnails close to the studied pixel are given a large weight, whereas pixels belonging to the thumbnails far from the studied pixel are assigned a low weight. This weighting is equivalent to applying a decreasing function of the distance (d) between the studied pixel and the center of the image (window) containing this pixel. For example:

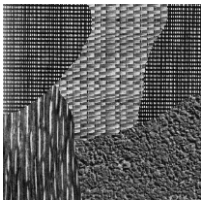
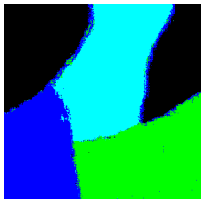
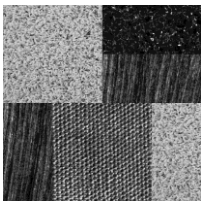
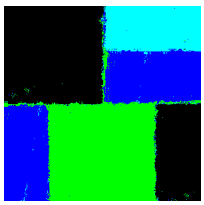
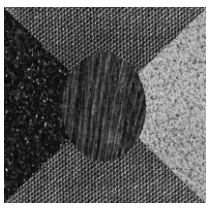
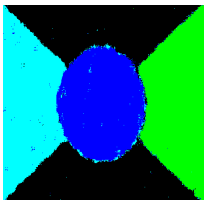
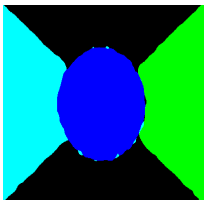
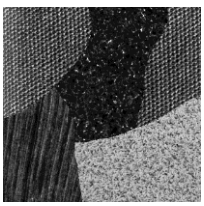
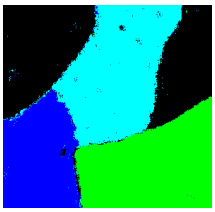
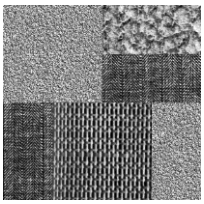
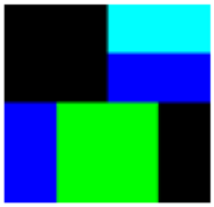
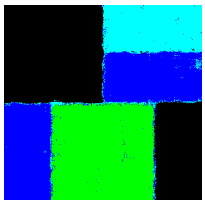
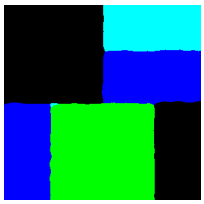
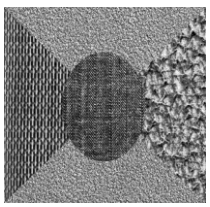
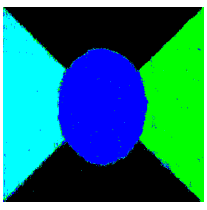
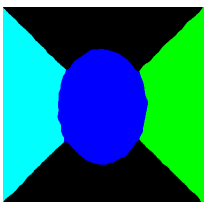
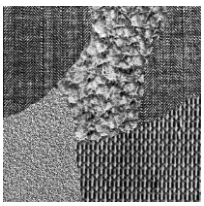
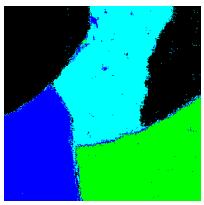
$$f(d) = e^{-d} \qquad IV.5$$

ou

$$f(d) = \frac{1}{1+d} \qquad IV.6$$

Table IV.3 Experimental results. The first column contains textured images in different shapes. The second column contains the ground truth map of different images. The third and fourth columns are the classification results using respectively the 1st fusion algorithm and the 2nd fusion algorithm

Original image	*Ground truth map*	*1st fusion algorithm*	*2nd fusion algorithm*
 ***(a1)***			
 ***(a2)***			
 ***(a3)***			
 ***(b1)***			
 ***(b2)***			

 *(b3)*			
 *(c1)*			
 *(c2)*			
 *(c3)*			
 *(d1)*			
 *(d2)*			
 *(d3)*			

The principle steps of the second fusion algorithm are:

Second fusion algorithm	
Step1: Extraction of features by DWT and Gabor filters.	
Step2: Neuronal classification of each estimated features vector. We obtain:	

Step3:	Scores1: for DWT.
	Scores2: for Gabor filters.
	while the recognition rate is changed **do**
	for each pixel **do**
	Bayesian fusion of Score1, 2 and the neighboring scores.
	end
	Calculate the recognition rate.
	end

The 4th column of table IV.3 corresponds to the classification results using this second fusion algorithm where it is very clear that the different textures are highly differentiated compared to the first fusion algorithm (3th column of table IV.3), and we notice the decrease of the error in the borders of different classes.

From the 3th column of table IV.4, we can see that by this second fusion algorithm, the quality of the segmented images is highly enhanced and the classification accuracies are augmented compared with the first fusion algorithm in all images.

We note that in each images (a), (b), (c), (d) we have tested the same textures in different shapes and we observe that our fusion methods give good classification results whatever the form of the used textures.

Table IV.4 Classification accuracy of the second fusion algorithm compared with the first one.

Image	*1st fusion algorithm*	*2nd fusion algorithm*
(a_1)	96.27 %	99.02 %
(a_2)	96.93 %	98.43 %
(a_3)	96.81 %	99.13 %
(b_1)	97.22 %	98.66 %
(b_2)	96.49 %	98.05 %
(b_3)	96.33 %	98.26 %
(c_1)	97.53 %	98.38 %
(c_2)	97.32 %	98.13 %
(c_3)	97.06 %	98.16 %
(d_1)	97.56 %	98.97 %
(d_2)	97.67 %	98.58 %
(d_3)	96.79 %	98.26 %

And in order to more evaluate this second fusion algorithm, we have used other evaluation criteria, because the main problem with classification accuracy is that it hides the detail you need to better understand the performance of the classification model.

As mentioned in section II.3.1, the confusion matrix shows the ways in which our classification model is confused when it makes predictions. It gives us insight not only into the errors being made by our classifier but more importantly the types of errors that are being made. It is this breakdown that overcomes the limitation of using classification accuracy alone.

Table IV.5 represents the confusion matrices of different classification results obtained by the second fusion algorithm, and table IV.6 represents the criteria: recall, precision and kappa obtained from these last confusion matrices.

As mentioned also in section II.3.3 and II.3.4, a high recall indicates that the class is correctly recognized and high precision indicates an example labeled as positive is indeed positive. So, as seen in table IV.6 we have obtained a high recall and high precision in all classes of different images, we have reached in some cases a 100% of recall or precision, so this second fusion algorithm gives a good recognition whatever the kind of texture.

As mentioned also in section II.3.6, the basic idea behind kappa is that some of the apparent classification accuracy could be due to chance. It is used to control only those instances that may have been correctly classified by chance. So, as seen in the eightieth column of table IV.6 we have obtained a kappa index between 0.97 and 0.99 in all cases, that means we have a near perfect agreement as mentioned in table II.3.

Table IV.5 Confusion matrices of different classification results.

Image	***Confusion Matrix***			
(a_1)	31991	0	278	213
	46	10033	87	115
	22	23	22155	1
	71	0	350	23419
(a_2)	44357	229	102	9
	3	25241	0	0
	25	105	24661	121
	302	0	38	25463

(a3)	$\begin{Bmatrix} 26946 & 22 & 7 & 11 \\ 192 & 23618 & 16 & 62 \\ 62 & 62 & 15972 & 20 \\ 82 & 46 & 10 & 21488 \end{Bmatrix}$
(b1)	$\begin{Bmatrix} 32443 & 20 & 0 & 19 \\ 72 & 10209 & 0 & 0 \\ 187 & 335 & 21496 & 183 \\ 352 & 0 & 27 & 23461 \end{Bmatrix}$
(b2)	$\begin{Bmatrix} 43111 & 688 & 61 & 837 \\ 0 & 25204 & 2 & 38 \\ 346 & 155 & 24253 & 158 \\ 7 & 39 & 0 & 25757 \end{Bmatrix}$
(b3)	$\begin{Bmatrix} 26445 & 33 & 23 & 485 \\ 43 & 23365 & 0 & 480 \\ 4 & 310 & 15700 & 102 \\ 0 & 0 & 5 & 21621 \end{Bmatrix}$
(c1)	$\begin{Bmatrix} 32364 & 6 & 15 & 97 \\ 79 & 10141 & 57 & 4 \\ 144 & 212 & 21188 & 657 \\ 257 & 0 & 0 & 23583 \end{Bmatrix}$
(c2)	$\begin{Bmatrix} 43720 & 313 & 0 & 664 \\ 17 & 25132 & 95 & 0 \\ 732 & 219 & 23779 & 182 \\ 11 & 2 & 0 & 25790 \end{Bmatrix}$
(c3)	$\begin{Bmatrix} 26428 & 355 & 0 & 203 \\ 43 & 23503 & 44 & 298 \\ 130 & 164 & 15607 & 215 \\ 0 & 0 & 0 & 21626 \end{Bmatrix}$

(d_1)	32470	0	12	0
	87	10191	3	0
	248	132	21821	0
	356	44	297	23143
(d_2)	44678	0	9	10
	670	24406	168	0
	387	3	24491	31
	323	0	83	25397
(d_3)	26550	211	221	4
	183	23531	170	4
	0	0	16116	0
	81	273	189	21083

Figure IV.7 and table IV.7 present the comparison results of the proposed fusion classification algorithms against other works from the literature; and they show clearly that the obtained classification rates outperform the results of other works. We have reached an accuracy of more than 98% in all cases, whereas the best result in the literature was less than 97% for all other works that used data fusion.

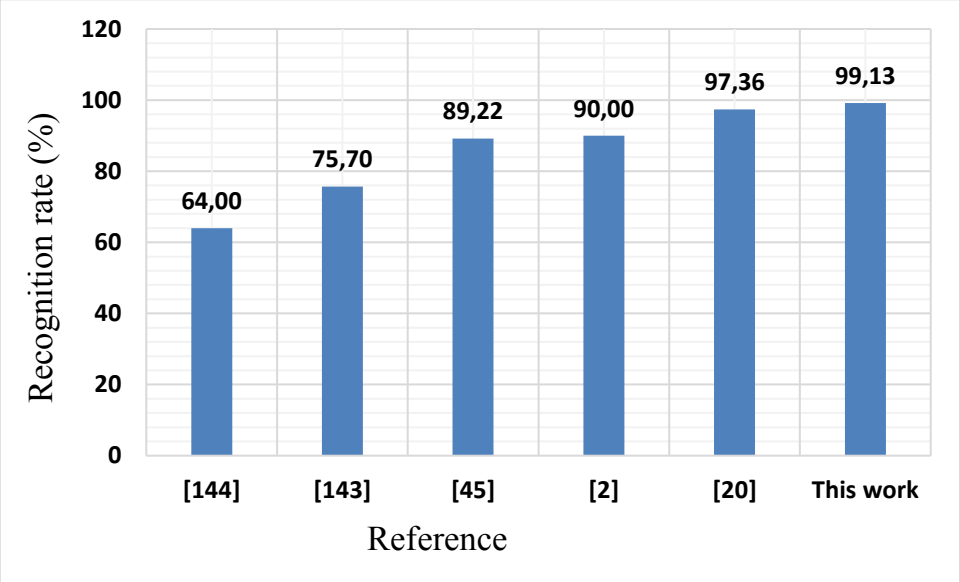


Figure IV.7 *Comparison results of textured image classification.*

Table IV.6 Criteria obtained from the confusion matrices.

Image		*Class 1*	*Class 2*	*Class 3*	*Class 4*		*Kappa*
(a1)	*Recall (%)*	*98.49*	*97.59*	*99.79*	*98.23*	*Overall Recall: 98.53*	*0.9811*
	Precision (%)	*99.57*	*99.77*	*96.87*	*98.61*	*Overall Precision: 98.71*	
(a2)	*Recall (%)*	*99.24*	*99.98*	*98.99*	*98.68*	*Overall Recall: 99.22*	*0.9894*
	Precision (%)	*99.26*	*98.69*	*99.43*	*99.49*	*Overall Precision: 99.22*	
(a3)	*Recall (%)*	*99.85*	*98.86*	*99.10*	*99.36*	*Overall Recall: 99.30*	*0.9910*
	Precision (%)	*98.77*	*99.45*	*99.79*	*99.57*	*Overall Precision: 99.40*	
(b1)	*Recall (%)*	*99.88*	*99.30*	*96.82*	*98.41*	*Overall Recall: 98.60*	*0.9813*
	Precision (%)	*98.15*	*96.64*	*99.87*	*99.15*	*Overall Precision: 98.45*	
(b2)	*Recall (%)*	*96.45*	*99.84*	*97.35*	*99.82*	*Overall Recall: 98.37*	*0.9736*
	Precision (%)	*99.19*	*96.62*	*99.74*	*96.14*	*Overall Precision: 97.92*	
(b3)	*Recall (%)*	*98.00*	*97.81*	*97.42*	*99.98*	*Overall Recall: 98.30*	*0.9774*
	Precision (%)	*99.82*	*98.55*	*99.82*	*95.30*	*Overall Precision: 98.37*	
(c1)	*Recall (%)*	*99.64*	*98.64*	*95.44*	*98.92*	*Overall Recall: 98.16*	*0.9760*
	Precision (%)	*98.54*	*97.90*	*99.66*	*96.89*	*Overall Precision: 98.25*	
(c2)	*Recall (%)*	*97.81*	*99.56*	*95.45*	*99.95*	*Overall Recall: 98.19*	*0.9747*
	Precision (%)	*98.29*	*97.92*	*99.60*	*96.82*	*Overall Precision: 98.16*	
(c3)	*Recall (%)*	*97.93*	*98.39*	*96.84*	*100.00*	*Overall Recall: 98.29*	*0.9779*
	Precision (%)	*99.35*	*97.84*	*99.72*	*96.80*	*Overall Precision: 98.43*	
(d1)	*Recall (%)*	*99.96*	*99.12*	*98.28*	*97.07*	*Overall Recall: 98.61*	*0.9815*
	Precision (%)	*97.91*	*98.30*	*98.59*	*100.00*	*Overall Precision: 98.70*	
(d2)	*Recall (%)*	*99.96*	*96.68*	*98.31*	*98.43*	*Overall Recall: 98.34*	*0.9808*
	Precision (%)	*97.00*	*99.99*	*98.95*	*99.84*	*Overall Precision: 98.94*	
(d3)	*Recall (%)*	*98.38*	*98.51*	*100.00*	*97.49*	*Overall Recall: 98.59*	*0.9797*
	Precision (%)	*99.02*	*97.98*	*96.53*	*99.96*	*Overall Precision: 98.37*	

Table IV.7 Comparative analysis of textured image classification.

Reference	Contribution	Recognition rate (%)
[144]	Combination of spectral and texture features using local spectral histogram	64.00
[143]	Combination of information from several Individual classifier in Multi class classification based on fuzzy theory	75.70
[45]	Fusion of classification results of sliding windows using all the windows containing the area to classify	89.22
[2]	Extraction of textures primitives with DWT to build an universal textons vocabulary that describes local features of textures	90.00
[20]	Combination of multiple decisions/scores provided by the same classifier over a sliding window.	97.36
This work	Bayesian fusion of multiple neural networks classifiers scores provided by the different features over a sliding window.	**Between 98.05 and 99.13**

The outcomes of our contributions demonstrate that it is possible to reach excellent fusion performance by neatly selecting the best fusion method. We also note that by our fusion methods, the segmentation results of the textured images are much improved and the majority of segmentation errors are situated in the borders of different classes. So, we will try, as future works, to use adaptive masks in order to more decrease the error at these borders.

IV.3. Application to mammographic images

Every year, 400 000 women die from breast cancer, but that rate can be reduced by 25% only with early detection. However, it takes about five years for a breast tumor to reach 1 mm, two years longer to reach 5mm and one or two years to measure 2 cm and be large enough to be detected by palpation [145 - 147].

Two of the most important mammographic indicators of breast cancer are masses (space occupying lesions) and microcalcifications (tiny flecks of calcium, like grains of salt). Breast

cancer studies have shown that early detection of these abnormalities boosts prognosis and reduces significantly the mortality rates.

Mammography is actually the best diagnostic mean for screening, and several images processing techniques have been used for mammograms interpretation in order to assist radiologists while detecting and/or identifying eventual abnormalities. On a mammogram, we can observe the breast, the pectoral muscle and the nipple, for oblique incidences (figure IV.8). A mammographic examination basically involves two incidences per breast. A woman will do many of these tests in her life.

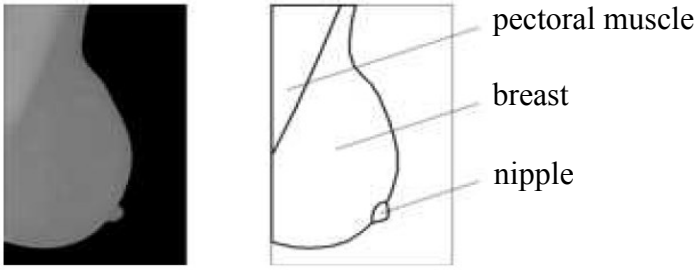


Figure IV.8 *breast Anatomy.*

In this thesis we only focus on one of breast cancer signs which are masses. Masses are space occupying lesions, seen on two different impacts. They are characterized by their shape (round, oval, lobulated, irregular), their contour (circumscribed, micro-lobulated, obscured, indistinct, speculated) and density (high, medium, low fat), as seen in figure IV.9). Breast cancers are never made of fat (radio-transparent) though they may trap grease. Lesions containing fat are: oil cysts, lipomas, the galactocele and mixed lesions (hamartoma). Mass containing fat is always benign [148].

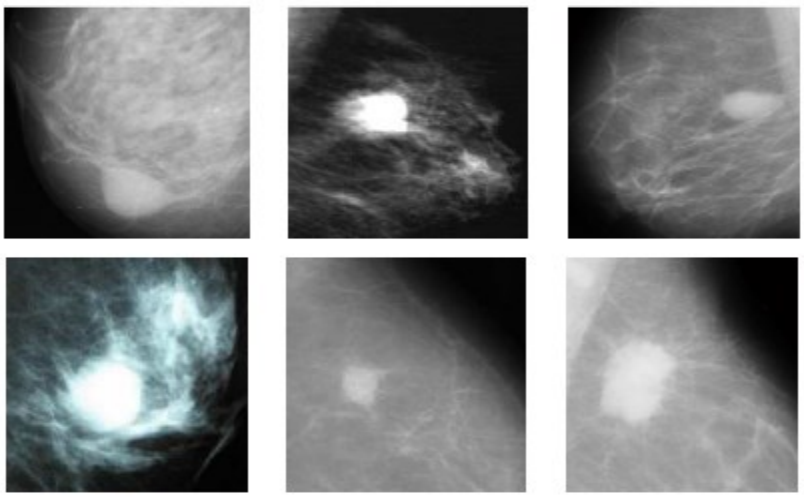


Figure IV.9 *Examples of masses.*

In [149], Cheng and al. gave a comparison of masses segmentation approaches, their advantages and drawbacks. Various works have been done to detect masses in mammography images using different methods. Sharma and Khanna [150] proposed a diagnostic aid system to detect abnormalities or suspicious areas in digital mammographic images and classify them as malignant or non-malignant using the descriptors of form and SVM classifier. Fatehia and Mawia [151] identified breast cancer using the texture and classified it as normal or abnormal, Jen and Yu [152] proposed a method for detecting abnormal mammograms by extracting some discriminant characteristics, first-order statistics, intensities and gradients. Kumar and Bandyopadhyay [153] used a multi-scale model form to locate pixels in the image, which can be part of a mass for identifying abnormal masses in the digital mammography images. Sampaio [154] detected the mammary masses using neural networks to segment the regions that might contain masses and SVM to classify the candidate regions as masses or non-masses.

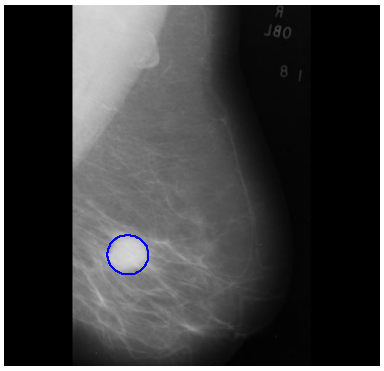
In this study we have used the MIAS database [155] containing 322 mammograms sized 1024 pixels x 1024 pixels. The images are arranged in pairs: those with even-number correspond to left MLO (medio-lateral oblique Mammograms) and those with odd-number are right MLO.

The mammograms contained in this base cover all diagnosis possibilities: healthy (208 images), with masses (56 images), with microcalcifications (25 images), with architectural distortions (18 images), or with asymmetries (15 images). In the case of the masses, the coordinates of the center of the anomaly and the radius corresponding to all the anomaly area were provided by radiologists (the mark of origin is the bottom left corner).

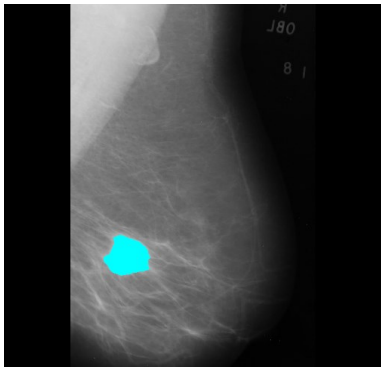
Our method consists on using our second fusion algorithm, seen previously, in detecting masses on MLO mammograms. The same procedure followed during the segmentation of textured images is applied in this context for the extraction of suspicious areas. For Gabor Filters, Wavelet transform, and Bayes fusion, the same parameters used in textured images segmentation, presented in section IV.2, have been maintained.

For the learning phase, we have used image mdb028 of the MIAS database, and to test our algorithm we have taken randomly MIAS images mdb025, mdb134, mdb184, and mdb271.

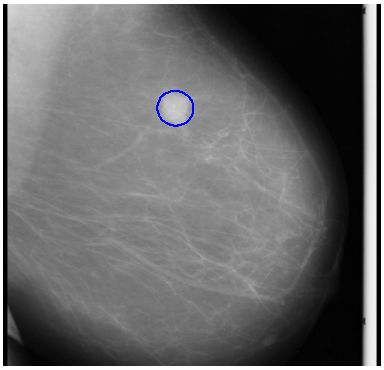
Figure IV.10 illustrates the obtained results (detected masses are displayed in cyan color) compared to expert decision (masses centers coordinates and radiuses shown in blue color).



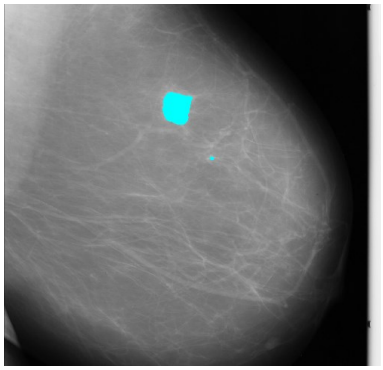
mdb028 (a)



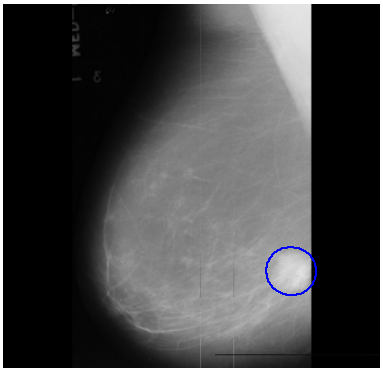
mdb028 (b)



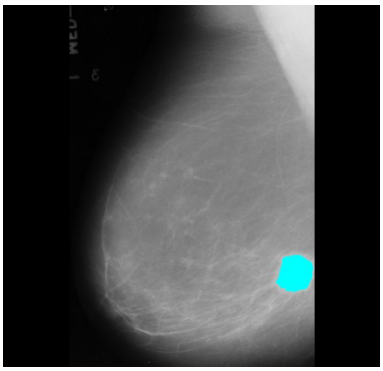
mdb134 (a)



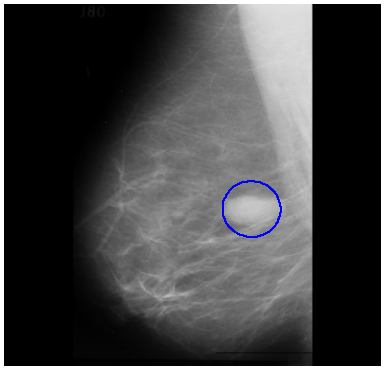
mdb134 (b)



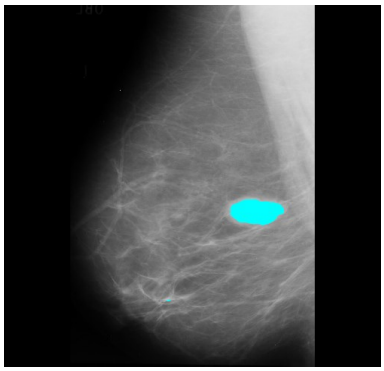
mdb271 (a)



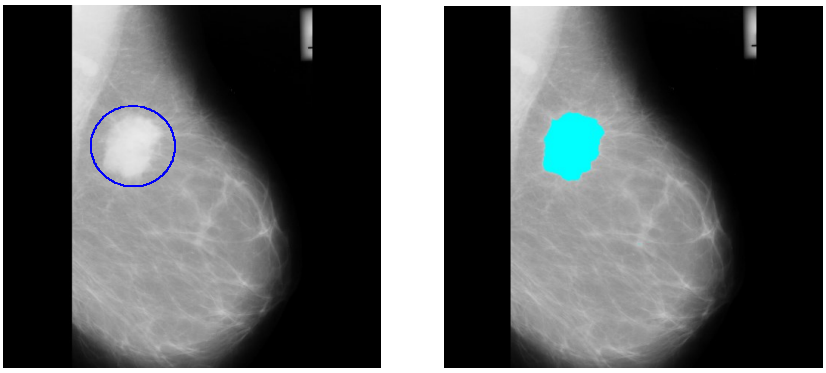
mdb271 (b)



mdb025 (a)



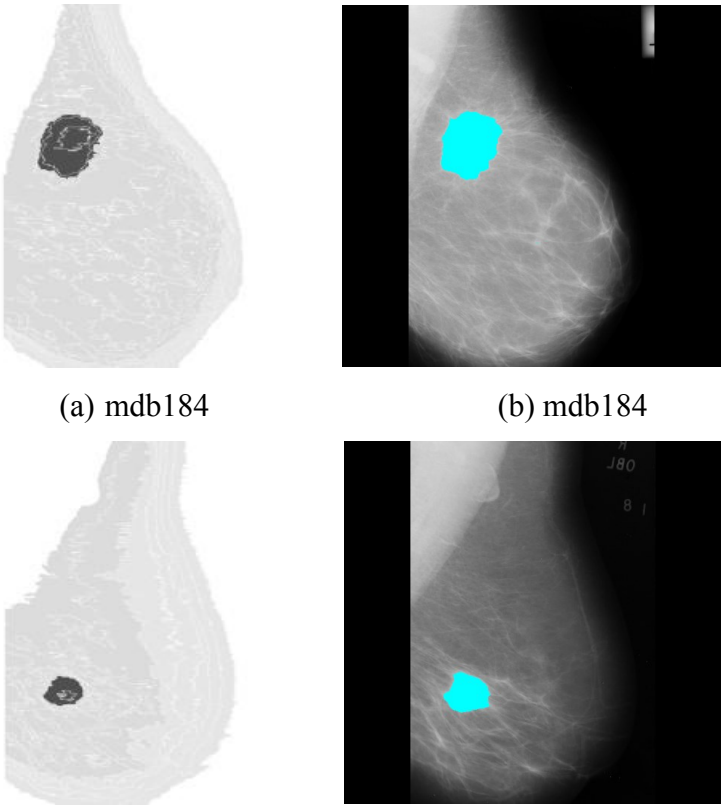
mdb025 (b)



mdb184 (a) mdb184 (b)

Figure IV.10 *Examples of results on MIAS database: (a) Original mammogram with expert masse location, (b) Masse segmentation using our approach.*

The evaluation of the used method will be based on a visual assessment, in terms of acceptable and unacceptable segmentation. So, as shown in figure IV.10, the results obtained on MIAS database are promising. Figure IV.11 shows a comparison of our proposed approach, on MIAS images mdb184 and mdb028, with another unsupervised technique proposed by Kanta Maitra et al [156], based on Divide and conquer algorithm.



(a) mdb184 (b) mdb184

(a) mdb028 (b) mdb028

Figure IV.11 *Comparison between results from: (a)* [156] *(b) Our approach.*

As seen in figure IV.11, the proposed approach has the advantage of being simple and precise. However, lack of ground truth prevents us from having accurate rate of our masses segmentation technique. So, we also need to build a new mammograms database with reliable and precise expert decision to be able to have perfect evaluation.

IV.4. Conclusion:

In this chapter we have presented all contributions made in this framework, and we have evaluated the performances of the proposed systems.

The proposed methodology is based on a chain consisting of several stages: calculation of textural descriptors; data classification; and fusion.

For synthetic images, we have proposed in the first time a segmentation that uses each texture descriptor individually. As a second approach, we integrated fusion before the decision-making. The scores obtained by the neural network have been fused using the probability theory. The obtained results show the interest of fusion compared to those obtained without fusion. Then, our fusion segmentation algorithm is applied on mammographic images to detect masses. The proposed system is tested on MIAS database.

The obtained results on synthetic and real images show the effectiveness of our algorithms for the segmentation of textured images.

GENERAL CONCLUSION AND PERSPECTIVES

Conclusion:

This thesis was mainly devoted to the study of the textured images segmentation based on the process of knowledge extraction from data, which has been adopted to answer our problematic.

We have carried out a study of the segmentation chain, from the parameters extraction phase to the evaluation phase of the results. For the characterization of textured images, we relied on textural information.

A multitude of texture analysis approaches exists in the literature; we have chosen the most used namely the cooccurrence matrices [12], Gabor filters [15], and wavelets [16].

In the first chapter, a panorama of different approaches of texture analysis, in a non-exhaustive way, is presented.

The second chapter was devoted to the study of textured image segmentation methods. Particular attention has been paid to the segmentation based on pixel's classification. We also presented the supervised and unsupervised classification methods and classification and segmentation evaluation approaches.

In chapter 3, the specificities of the information fusion and its application for the textured images segmentation were first described. After a general description of information fusion in image processing, we presented the two main methods of information fusion namely the probabilistic approaches that are the most prevalent and the fuzzy set theory.

In chapter 4 we have presented all contributions made in this framework, and we have evaluated the performances of the proposed systems. We started by extracting the features using GLCM matrices, Gabor filters and the wavelet transform from textured images. After that, each estimated vector of features for every pixel was sent to the neural networks classifier for primary labeling. Next, as a first fusion scheme, for each pixel a Bayes fusion method was used to combine the three scores obtained by the neural networks. Then a new fusion model for improving decision-making was used, it consists of combining the scores of each pixel within a sliding window. The proposed fusion algorithms were tested on synthetic images from Brodatz and DTD datasets; and they provided more information and improved classification results. Consequently, it is possible to attain an excellent fusion performance by properly selecting the perfect fusion method.

We subsequently introduced a computer-assisted diagnostic system for mammographic images. The main objective of this system is to detect the presence of masses in mammography images.

Perspectives:

Concerning future works, several perspectives are offered to the work carried out.

About the segmentation of textured images, the first axis is the use of an adaptive mask in features extraction, in order to more decrease the error in the borders of different classes; use of other fusion methods like belief theory; comparison with other algorithms published in the literature; use of other real databases to confirm the robustness of different approaches developed in this study.

For the diagnostic help system of mammography images, we will deepen the knowledge related to this system especially the evaluation of the segmentation stage, as well as the validation of the proposed approach on the whole database, specifically on dense tissue, and on another databases. Finally, complete the system by classifying abnormal mammographic images into benign and malignant.

Publications

- A. Zitouni, F. Benkouider, F. Chouireb, M. Belkheiri, “ Classification of Textured Images Based on New Information Fusion Methods ”, IET Image Processing, vol.13, issue 9, pp. 1540 - 1549, July 2019.

  DOI: 10.1049/iet-ipr.2018.6256.

- A. Zitouni, F. Benkouider, F. Chouireb, M. Belkheiri, “ Comparison Between Gabor Filters and Wavelets Transform for Classification of Textured Images ”, The 4th International Conference on Electrical Engineering and Control Applications - ICEECA'2019 December 17-19th, 2019, Constantine, Algeria.

REFERENCES

[1] Sahbani, M.H., Hamrouni, K. : 'Segmentation d'images texturées par transformée en ondelettes et classification C-moyenne floue'. Proc. 3rd Int. Conf. Sciences of Electronic, Technologies of Information and Telecommunications, Tunisia, March 27-31, 2005.

[2] Kennel, P., Fiorio, C. et Borne, F. (2015): 'Supervised image segmentation using q-shift dual-tree complex wavelet transform coefficients with a texton approach'. *Pattern Analysis and Applications*, pages 1-11.

[3] Karine, A., Lasmar, N., Baussard, A. et El Hassouni, M. (2015): 'Sonar image segmentation based on statistical modeling of wavelet sub bands'. ACS/IEEE International Conference on Computer Systems and Applications AICCSA.

[4] Wu, H. et Li, M. e. Z. M. e. Z. J. e. S. J. (2013) : 'Texture segmentation via scattering transform'. *International Journal of Signal Processing, Image Processing and Pattern Recognition*, 6(2):165-174.

[5] Wang, B. et Zhang, L. (2003): 'Supervised texture segmentation using wavelet transform'. In Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, pages 1078-1082.

[6] Wild, P., Hofbauer, H., Ferryman, J., et al.: '*Robust iris image segmentation*', in Rathgeb, C. and Busch, C. (eds.): 'Iris and periocular biometric recognition' (IET Book Series on Advances in Biometrics. The Institution of Engineering and Technology, London, 2017), pp. 57-82.

[7] Peña-Barragán, JM., Ngugi, MK., Plant, RE.: 'Object-based crop identification using multiple vegetation indices, textural features and crop phenology', *Remote Sens Environ*, 2011, 115, (6), pp. 1301–1316.

[8] Sasikala, M., Kumaravel, N., Subhashini, L.: 'Automatic Tumor Segmentation using Optimal Texture Features'. IET 3rd Int. Conf. On Advances in Medical, Signal and Information Processing, MEDSIP, 2006, pp. 1-4.

[9] Filiberto, P., Gema, G., Pedro, G.S., *et al*.: 'Multi-spectral texture characterization for remote sensing image segmentation'. Proc. 4th Iberian Pattern Recognition and Image Analysis Conference, 2009, pp. 257–264.

[10] Pu, H., Sun, D-W., Ma, J., *et al*.: 'Using wavelet textural features of visible and near infrared hyperspectral image to differentiate between fresh and frozen-thawed pork', *Food Bioprocess Tech*, 2014, 7, (11), pp.3088-3099.

[11] Su, H., Sheng, Y., Du, P., *et al*.: 'Hyperspectral image classification based on volumetric texture and dimensionality reduction', *Frontiers of Earth Science*, 2015, 9, (2), pp. 225–236.

[12] Haralick, R. (1979): 'Statistical and structural approaches to texture'. In Proceedings of the IEEE, volume 67, pages 786-804.

[13] Galloway, M. (1975): 'Texture analysis using gray level run lengths'. *Computer Graphics and Image Processing*, 4:172-179.

[14] Ojala, T. et Pietikainen, M. e. H. D. (1996): 'A comparative study of texture measures with classification based on featured distributions'. *Pattern Recognition*, 29:51-59.

[15] Gabor, D. (1941): 'Theory of communication'. In J.IEE, volume 93, pages 429-457.

[16] Mallat, S. (1989): 'A theory of multiresolution signal decomposition: the wavelet representation'. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674-693.

[17] Mallat, S. (2012): 'Group invariant scattering'. *Communication on Pure and Applied Mathematics*, 65(10):1331-1398.

[18] Iftene, T., Safia, A. : 'Comparaison Entre La Matrice De Cooccurrence Et La Transformation En Ondelettes Pour La Classification Texturale Des Images Hrv (Xs) De Spot', *Télédétection*, 2004, 4, (1), pp. 39–49.

[19] Iggane, M., Mammass, D., Khenchaf, A., *et al*. : 'Segmentation d'images sonar par matrice de co-occurrence'. Proc. 5th Int. Conf. Sciences of Electronic, Technologies of Information and Telecommunications, Tunisia, March 22-26, 2009.

[20] Anibou, Ch., Saidi, M.N., Aboutajdine, D.: 'Classification of Textured Images Based on Discrete Wavelet Transform and Information Fusion', *J. Inf. Process. Syst*, 2015, 11, (3), pp. 421-437.

[21] Cover, T. M. et Hart, P. (1967): 'Nearest neighbor pattern classification'. *IEEE Transactions on information theory*, 13:21-27.

[22] Vapnik, V. N. (1998): 'Statistical learning theory'. John Wiley and Sons.

[23] Abdi, H. (1994). 'Neural network primer'. *Journal of Biological Systems*, 2:247-281.

[24] Thangarajah, A., Qingming, JW., Hui, Z.: 'Effect of Fusing Features from Multiple DCNN Architectures in Image Classification', *IET Image Processing*, July 2018, 12, (7), pp. 1102 – 1110.

[25] Bahrampour, S., Nasrabadi, N.M., Ray, A., *et al*.: 'Multimodal task-driven dictionary learning for image classification', *IEEE Trans. Image Process*., 2016, 25, pp. 24–38.

[26] Akilan, T., Jonathan Wu, Q.M., Safaei, A., *et al*.: 'A late fusion approach for harnessing multi-CNN model high-level features'. IEEE Int. Conf. Systems, Man, and Cybernetics, SMC, 2017, pp. 566–571.

[27] Cai, S., Zhang, L., Zuo, W., *et al*.: 'A probabilistic collaborative representation based approach for pattern classification'. Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2950–2959.

[28] Chen, S., Yang, J., Luo, L., *et al*.: 'Low-rank latent pattern approximation with applications to robust image classification', *IEEE Trans. Image Process.*, 2017, 26, (11), pp. 5519–5530.

[29] Fernando, B., Fromont, E., Muselet, D., *et al*.: 'Discriminative feature fusion for image classification'. Int. Conf. Pattern Recognition (ICPR), 2012, pp. 3434–3441.

[30] Hoashi, H., Joutou, T., Yanai, K.: 'Image recognition of 85 food categories by feature fusion'. Proc. Second Workshop on Multimedia for Cooking and Eating Activities, December 2010, p. 296-301.

[31] Park, D.-C.: 'Multiple feature-based classifier and its application to image classification'. Proc. IEEE Int. Conf. Data Mining Workshops, 2010, pp. 65–71.

[32] Gehler, P., Nowozin, S.: 'On feature combination for multiclass object classification'. Proc. IEEE 12th Int. Conf. Computer Vision (ICCV), 2009.

[33] Khan, F., van de Weijer, J., Vanrell, M.: 'Modulating shape features by color attention for object recognition', *Int. J. Comput. Vis.*, 2012, 98, pp. 49–64.

[34] Ai, D., Duan, G., Han, X., *et al*.: 'Multiple feature and fusion based on generalized n-dimensional independent component analysis'. Proc. Int. Conf. Pattern Recognition (ICPR), 2012, 21, pp. 971–974.

[35] Liang, X., Huang, D.S.: 'Image segmentation fusion using weakly supervised trace-norm multi-task learning method', *IET Image Processing*, 2018, 12, (7), pp. 1079 – 1085.

[36] Bloch, I.: 'Information Fusion in Signal and Image Processing', (Hoboken, NJ: John Wiley & Sons, 2008).

[37] Hérault, J., Jutten, Ch. : 'Réseaux neuronaux et traitement du signal', (Hermès, 1994).

[38] Jodouin, J. F. : 'Les réseaux de neurones. Principes et définitions', (Hermès, 1994).

[39] Rosenfeld, A. (1976): 'Digital Picture Analysis', volume 11. Springer Berlin Heidelberg.

[40] Haralick, R. et Shanmugam, K. e. D. I. (1973): 'Textural features for image classification'. In Systems, Man and Cybernetics, IEEE Transactions on, pages 610-621.

[41] Lasmar, N. (2012) : 'Modélisation stochastique pour l'analyse d'images texturées : Approches Bayésiennes pour la caractérisation dans le domaine des transformées'. Thèse de doctorat, Univ. Bordeaux.

[42] Materka, A. et Strzelcki, M. (1998): 'Texture analysis methods: A review'. Technical university of Lodz, institute of electronics, COST B11 report, Brussels, pages 9-11.

[43] Mirzapour, F., Ghassemian, H.: 'Fast GLCM and Gabor Filters for Texture Classification of Very High-Resolution Remote Sensing Images', *Int. J. Inf. Commun. Technol. Res.*, 2015, 7, (3), pp. 21-30.

[44] Mirzapour, F., Ghassemian, H.: 'Using GLCM and Gabor filters for classification of PAN images'. Proc. Iranian Conf. Electrical Engineering, 2013, pp. 1-6.

[45] Laanaya, H., Martin, A., Aboutajdine, D., *et al*.: 'Classifier fusion for post-classification of textured images'. Proc. 11th International Conf. Information Fusion, Cologne, Germany, 2008, pp. 1-7.

[46] Shangguan, W. et Hao, Y. e. a. (2008): 'The research of satellite cloud image recognition base on variational method and texture feature analysis'. In IEEE Conference on Industrial Electronics and Applications, pages 2816-2820.

[47] Mehri, M., Mhiri, M. and Gomez-Kramer, P., Héroux, P., Mahjoub, M. et al. (2014) : 'Etude comparative de trois ensembles de descripteurs de texture pour la segmentation de documents anciens'. In Treizième Colloque International Francophone sur l'Ecrit et le Document (CIFED), pages 41-56.

[48] Laanaya, H. (2007) : 'Classification en Environnement Incertain : Application à la Caractérisation de Sédiments Marins'. Thèse de doctorat, Université de Bretagne Occidentale, ENSIETA, Brest.

[49] Amroun, f. (2013) : 'Extraction de la composante texturée d'une image'. Mémoire de D.E.A., Université Mouloud Mammeri, Tizi-Ouzou.

[50] Akono, A. et Tonye, E. e. N. A. (2006) : 'Etude comparative de paramètres de texture d'ordre deux et trois pour une classification d'images RSO dans un contexte de multi résolution hiérarchique'. *Télédétection*, 6(4) :249-261.

[51] Randen, T. et Husoy, J. (1999): 'Filtering for texture classification: comparative study'. *Pattern Analysis and Machine Intelligence*, IEEE Transactions on, 21 : 291-310.

[52] Maenpaa, T. et Pietikainen, M. (2004): 'Texture analysis with local binary patterns', chapitre 1, pages 1-20. WSPC.

[53] Chaimae ANIBOU : 'Segmentation d'Images Texturées par Classification et Fusion d'Informations : Application aux Images Satellitaires et Médicales'. PhD. Dissertation, Université Mohammed V, Faculté des Sciences, Rabat, Maroc, 2016.

[54] Sivalingamaiah, M., Reddy, B. D. V.: 'Texture segmentation using multichannel Gabor filtering', *IOSR Journal of Electronics and Communication Engineering*, 2012, 2, pp. 22-26.

[55] Bosnjak, A., Montilla, G., Torrealba, V.: 'Medical Images Segmentation using Gabor Filters applied to Echocardiographic Images', *Computers in Cardiology, IEEE*, 1998, 25, pp. 457-460.

[56] Jain, A.K., Farrokhnia, F.: 'Unsupervised texture segmentation using Gabor filters', *Pattern Recognition*, 1991, 24, pp. 1167-1186.

[57] Grossman, A., Morlet, J.: 'Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape', *SIAM Journal on Mathematical Analysis*, 1984, 15, (4), pp. 723-736.

[58] Scheunders, P., Livens, S., Van de Wouwer, G., *et al.*: 'Wavelet-based Texture Analysis', *Int. J. Computer Science and Information Management*, December 1997.

[59] Arivazhagan, S., Ganesan, L.: 'Texture segmentation using wavelet transform', *Pattern Recognition Letters*, 2003, 24, (16), pp. 3197–3203.

[60] Arivazhagan, S. et Ganesan, L. (2003) : 'Texture segmentation using wavelet transform'. *Pattern Recognition Letters*, 24 :3197-3203.

[61] Beucher, S. (1990) : 'Segmentation d'images et morphologie mathématique'. Thèse de doctorat, Ecole Nationale Supérieure des Mines de Paris.

[62] Kato, Z. et Pong, T. (2006): 'A markov random field image segmentation model for color textured images'. *Image and Vision Computing*, 24(10):1103-1114.

[63] Hsiao, J. et Sawchuk, A. (1989): 'Supervised textured image segmentation using feature smoothing and probabilistic relaxation techniques'. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(12): 1279-1292.

[64] Rosenbergern, C. (2007) : 'Mise en œuvre d'un Système Adaptatif de Segmentation d'Images'. Thèse de Doctorat, Université de Rennes 1, décembre 1999.

[65] Kass, M., Witkin, A. et Terzopoulos, D. (1988) : 'Snakes : actif contour models'. *Computer vision, Graphics and image processing*, pages 321-331.

[66] Rousselle, J. (2003) : 'Les contours actifs, une méthode de Segmentation : Application à L'Imagerie médicale'. Thèse de doctorat, Université François Rabelais de Tours, 9 juillet 2003.

[67] Adams, R. et Bischof, L. (1994) : 'Snakes : actif contour models'. IEEE Transactions on Pattern Analysis and Machine Intelligence, pages 641-647.

[68] Deng, Y. and Manjunath, S. B. (2001): 'Unsupervised segmentation of color-texture regions in images and video'. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23:800-810.

[69] Fung, P., Grebbin, G. et Attikiouzel, Y. (1990): 'Model-based region growing segmentation of textured images'. In International Conference on Acoustics, Speech, and Signal Processing, volume 4, pages 2313-2316.

[70] Lira, J. et Frulla, L. (1998): 'An automated region growing algorithm for segmentation of texture regions in sar image'. *int. j. remote sensing*, 19(18):3595-3606.

[71] Anibou, C., Saidi, M. et Aboutajdine, D. (2015b): 'Computer aid diagnostic in mammogram image using Susan algorithm and hierarchical watershed transform'. In International Symposium on Ubiquitous Networking, UNet, volume 366, pages 355-366. Springer Verlag.

[72] Samet, H. (1984): 'The quadtree and related hierarchical data structures'. *ACM Computing Surveys*, 16:187-260.

[73] R.S., J. et Chang, S. (1994): 'Qyad-tree segmentation for texture-based image query'. In ACM 2nd International Conference on Multimedia.

[74] Kermad, C., Chehdi, K. et Cariou, C. (1995) : 'Segmentation d'images par multi-seuillage et fusions de régions labelisées minimisant un critère de similarité'. In 15 colloque GRESTI – Juan–Les–Pins, du 18 au 21 Septembre 1995.

[75] Saidi, M. (2010) : 'Reconnaissance de Formes et d'Objets en Environnement Incertain : Application à la Reconnaissance de Cibles Radar'. Thèse de doctorat, Université Mohammed V-Agdal.

[76] Celeux G,Diday E., Govaert G., Lechevallier Y., Ralam-Bondrainy H. : 'Classification Automatique des Données'. Bordas, Paris, 1989.

[77] BOULFANI Yasmine, DOUMANDJI Samah : 'Implémentation sur DSP TMS320C5000 de filtres optimaux appliqués aux images et introduction de réseaux neuronaux', Mémoire de fin d'études, Ecole Nationale Polytechnique, Alger, Algérie, juin 2004.

[78] BOULEMNADJEL Amel : 'Partitionnement neuronal et validité des classes Application à la segmentation d'images', Mémoire de Magister, Université Mentouri – Constantine Le 07/07/2009.

[79] http://gsite.univprovence.fr/gsite/Local/umr_6149/umr/page_perso/Touzet/Les réseaux de neurones artificiels.pdf.

[80] Jason Brownlee, : 'What is a Confusion Matrix in Machine Learning', Available online at https://machinelearningmastery.com/confusion-matrix-machine-learning/

[81] Ting K.M. (2017) : 'Confusion Matrix'. In: Sammut C., Webb G.I. (eds) Encyclopedia of Machine Learning and Data Mining. Springer, Boston, MA.

[82] J Cohen. 'A coefficient of agreement for nominal scales'. *Educational& Psychological Measurement*, 20:37–46, 1960. 13, 24.

[83] J Cohen: 'Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit'. *Psychological Bulletin*, 70:426–443, 1968. 11, 13, 19.

[84] J L Fleiss, J Cohen, and B S Everitt: 'Large sample standard errors of kappa and weighted kappa'. *Psychological Bulletin*, 72(5):323–327,1969. 13, 14, 19.

[85] LANDIS JR, KOCH GG: 'The measurement of observer agreement for categorical data'. *Biometrics* 1977a; 33: 159-174.

[86] LAPLANCHE A, COM-NOUGUE C, FLAMANT R : 'Méthodes statistiques appliquées à la recherche clinique'. Flammarion Médecine-Sciences, Paris, 1987.

[87] Bloch, I. et Maitre, H. (1994): 'Data fusion in image processing: information models and decisions'. *Traitement du Signal*, 11(6).

[88] Martin, A. (2012) : 'Fusion de classifieurs pour la classification d'images sonar'. *Revue Nationale des Technologies de l'Information*, E (5), pp.259-268.

[89] Chitroub, S. (2004) : 'Combinaison de classifieurs : Une approche pour l'amélioration de la classification d'images multi sources/multi dates de télédétection'. *Télédétection*, 4(3) :289-301.

[90] Tabassian, M., Ghaderi, R. et Ebrahimpour, R. (2012): 'Combination of multiple diverse classifiers using belief functions for handling data with imperfect labels'. *Expert Systems with Applications*, 39:1698-1707.

[91] Bloch, I. (2003): 'Information Fusion in Signal and Image Processing'. John Wiley and Sons.

[92] B.V. Dasarathy: 'Sensor Fusion Potential Exploitation - Innovative Architectures and Illustrative Applications'. *Proceeding of the IEEE*, 85(1):2438, Janvier 1997.

[93] P. Verlinde: 'A Contribution to Multi-Modal Identity Verification using Decision Fusion'. Thèse de doctorat, École Nationale Supérieure des Télécommunications, Paris, France, 1999.

[94] D. Dubois et H. Prade : 'La fusion d'informations imprécises'. *Traitement du Signal*, 11(6) :447-458, 1994.

[95] D. Dubois et H. Prade: 'Possibility theory and data fusion in poorly informed environments'. *IFAC*, 2(5) :811-823, 1994.

[96] I. Bloch : 'Incertitude, imprécision et additivité en fusion de données : point de vue historique'. *Traitement du Signal*, 13(4) :267-288, 1996.

[97] Ph. Smets: 'Imperfect information: Imprecision - Uncertainty'. In A. Motro et Ph. Smets, éditeurs: Uncertainty Management in Information Systems, pages 225-254. Kluwer Academic Publishers, 1997.

[98] Ph. Smets: 'The variety of ignorance and the need for well-founded theories'. *Information Sciences*, 57-58:135-144, 1991.

[99] I. Bloch: 'Information Combination Operators for Data Fusion: A Comparative Review with Classification'. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 26(1):52-67, Janvier 1996.

[100] P.K. Varshney: 'Special Issue on Data Fusion'. *Proceeding of the IEEE*, 85(1):3-5, Janvier 1997.

[101] M. Delplanque, A.M. Desolt-Jolly et J. Jamin : 'Fusion dissymétrique d'informations incomplètes pour la classification d'objets sous-marins'. *Traitement du Signal*, 14(5) :511-522, 1997.

[102] F. Janez et A. Appriou : 'Théorie de l'Evidence et cadres de discernement non exhaustifs'. *Traitement du Signal*, 13(3) :237-250, 1996.

[103] F. Janez et A. Appriou: 'Theory of evidence and non-exhaustive frames of discernment: Plausibilities correction methods'. *International Journal of Approximate Reasoning*, 18 :1-19, 1998.

[104] A. Appriou : 'Décision et Reconnaissance des formes en signal', chapitre Discrimination multi signal par la théorie de l'évidence, pages 219-258. Hermes Science Publication, 2002.

[105] F. Janez : 'Fusion de sources d'information définies sur des référentiels non exhaustifs différents'. Thèse de doctorat, Université d'Angers, Novembre 1996.

[106] F. Delmotte et P. Borne: 'Modeling of Reliability with Possibility Theory'. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, 28(1):78-88, Janvier 1998.

[107] A. Josang: 'The Consensus Operator for Combining Beliefs'. *Artificial Intelligence Journal*, 141(1-2):157-170, 2002.

[108] E. Lefevre, O. Colot et P. Vannoorenberghe: 'Belief function combination and conflict management'. *Information Fusion*, 3:149-162, 2002.

[109] D. Dubois and H. Prade. 'Possibility Theory'. Plenum Press, New-York, 1988.

[110] C.E. Shannon: 'A mathematical theory of communication'. *Bell System Technical Journal*, 27:379-423, October 1948.

[111] Martin, A. (2005) : 'La fusion d'informations'. Polycopié de cours ENSIETA.

[112] L. Xu, A. Krzyzak et C.Y. Suen: 'Methods of Combining Multiple Classifiers and Their Application to Handwriting Recognition'. IEEE Transactions on Systems, Man Cybernetics, 22(3):418-435, Mai 1992.

[113] A. P. Dempster: 'Upper and lower probabilities induced by a multivalued mapping'. The annals of mathematical statistics, pp. 325-339, 1967.

[114] G. Shafer: 'A mathematical theory of evidence', vol. 1: Princeton university press Princeton, 1976.

[115] P. Smets: 'The combination of evidence in the transferable belief model', *IEEE Transactions on pattern analysis and machine intelligence*, vol. 12, pp. 447-458, 1990.

[116] Lotfi Askar Zadeh: 'Fuzzy algorithms'. *Information and control*, 12 :94–102, 1968.

[117] Didier Dubois and Henri Prade : 'Théorie des possibilités : Applications à la Représentation des Connaissances en Informatique'. Masson, Paris. IMAGE EVALUATION TEST TARGET, 1985.

[118] Mohammad Homam Alsun, Laurent Lecornu, Basel Solaiman, Clara Le Guillou, and Jean Michel Cauvin : 'Medical diagnosis by possibilistic classification reasoning'. In Information Fusion (FUSION), 2010 13th Conference on, pages 1–7. IEEE, 2010.

[119] D. Dubois and H. Prade: 'Possibility theory and its applications: a retrospective and prospective view'. In Fuzzy Systems, 2003. FUZZ'03. The 12th IEEE International Conference on, volume 1, pages 5–11. IEEE, 2003.

[120] Jérémy Rohmer : 'La théorie des possibilités comme outil de représentation des incertitudes épistémiques d'une chaîne de traitement du risque sismique'. Proceedings AFPS 2007, 2007.

[121] S. Deveughele and B. Dubuisson : 'Adaptabilité et combinaison possibiliste : application à la vision multi-caméras'. *TS. Traitement du signal*, 11(6) :559–568, 1994.

[122] Didier Dubois, Laurent Foulloy, Gilles Mauris, and Henri Prade : 'Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities'. *Reliable computing*,10(4) :273–297, 2004.

[123] Didier Dubois and Henri Prade: 'Additions of interactive fuzzy numbers'. Automatic Control, IEEE Transactions on, 26(4) :926–936, 1981.

[124] Isabelle Bloch: 'Estimation of class membership functions for gray-level based image fusion'. *Image Processing*, 1997. Proceedings., 3 :268–271, 1997.

[125] Weibei Dou : 'Segmentation d'images multispectrales basée sur la fusion d'informations : application aux images IRM'. PhD thesis, Université de Caen, 2006.

[126] Enric Plaza, Claudi Alsina, R López de Mántaras, J Aguilar, and Jaume Agusti: 'Consensus and knowledge acquisition'. In Uncertainty in Knowledge-Based Systems, pages 294–306. Springer, 1987.

[127] Isabelle Bloch and Henri Maître : 'Les méthodes de raisonnement dans les images'. Ecole Nationale Supérieure des Télécommunications-CNRS UMR, 5141, 2004.

[128] Arnaud. M : 'La fusion d'informations'. Polycopié de cours ENSIETA-Réf,1484 :117, 2005.

[129] Clark J.J., Yuille A.L.: 'Data Fusion for Sensory Information Processing Systems', *Kluwer Academic Publishers*, Boston, 1990.

[130] Chauvin S. : 'Evaluation des performances du modèle bayésien de fusion appliqué à l'imagerie satellitaire', 15° colloque GRETSI, Juan les Pins, Septembre 1995, pp 949-952.

[131] Marzouki A. : 'Segmentation statistique d'images radar', Thèse de l'Université des Sciences et Technologies de Lille. Novembre 1996, 156 pages.

[132] Dromigny-Badin, A. : 'Fusion d'images par la théorie de l'évidence en vue d'applications médicales et industrielles'. PhD. Dissertation, Institut National des Sciences Appliquées de Lyon, 1998.

[133] L.I. Kuncheva: 'Switching Between Selection and Fusion in Combining Classifiers: An Experiment'. IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, 32(2):146-156, Avril 2002.

[134] L.I. Kuncheva, M. Skurichina et R.P.W. Duin: 'An experimental study on diversity for bagging and boosting with linear classifiers'. *Information Fusion*, 3(4): 245-258, Décembre 2002.

[135] S. Chauvin : 'évaluation des performances du modèle Bayésien de fusion appliqué à l'imagerie satellitaire'. In GRETSI, Juan-les-Pins, France, Septembre 1995.

[136] Castanedo, F. (2013): 'A Review of Data Fusion Techniques', *The Scientific World Journal*, Volume 2013, Article ID 704504,19 pageshttp://dx.doi.org/10.1155/ 2013 /704504

[137] A. A. Aguileta ,R. F. Brena ,O. Mayora ,E. M.Minero-Re and L. A. Trejo: 'Multi-Sensor Fusion for Activity Recognition—A Survey', *Sensors* 2019, 19(17), 3808 ; https://doi.org/10.3390/s19173808 - 03 Sep 2019

[138] Wierenga, B. et Kluytmans, J. (1994): 'Neural nets versus marketing models in time series analysis: a simulation studies'. In 23 annual conference, European marketing association, Maastricht, pages 1139-1153.

[139] Venugopal, V. et Baets, W. (1994): 'Neural networks and statistical techniques in marketing research: A conceptual comparison'. *Pattern Recognition*, 12:30-38.

[140] Shepard, D. (1990): 'The new direct marketing', chapter Business one Irwin Homewood IL.

[141] Cimpoi, M., Maji, S., Kokkinos, I., et al.: 'Describing Textures in the Wild'. Proc. IEEE Conf. Computer Vision and Pattern Recognition, Columbus, OH, 2014, pp. 3606-3613.

[142] Cimpoi, M., Maji, S., Vedaldi, A.: 'Deep filter banks for texture recognition and segmentation'. Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Boston, MA,2015, pp.3828-3836.

[143] Fauvel, M., Chanussot, J., Benediktsson, J.: 'Decision fusion for the classification of urban remote sensing images', IEEE Transactions on Geoscience and Remote Sensing, 2006, 44, (10), pp. 2828-2838.

[144] Yuan, J., Wang, D., Li, R.: 'Remote sensing image segmentation by combining spectral and texture features', IEEE Transactions on Geosciences and Remote Sensing, 2014, 52, (1), pp. 16-24.

[145] http://sante-medecine.commentcamarche.net/contents/cancer/13_le-cancer-du sein. ph p3 #les statistiques-alarmantes-du-cancer-du-sein

[146] http://www.cancerscreening.nhs.uk/breastscreen/publications/breast-screeningfrench .pdf

[147] http://www.nhs.uk/translationfrench/Documents/Cancer of the breast female French FINAL.pdf.

[148] H. Boulehmi, H. Mahersia, K. Hamrouni, S. Boussetta, N. Mnif: 'Breast cancer detection: a review on mammograms analysis techniques', International multi-conference on Signals, Systems and Devices – SSD'13, Hammamet – Tunisia, march 18-21, 2013.

[149] H.D. Cheng, X.J. Shi, R. Min, L.M. Hu, X.P. Cai, H.N. Du: 'Approaches for automated detection and classification of masses in mammograms', *Pattern Recognition* 39 (2006) 646 – 668, 2006.

[150] Sharma.S et Khanna.P (2015): 'Computer-aided diagnosis of malignant mammograms using Zernike moments and SVM'. *Journal of Digit Imaging*, 28(1):77-90.

[151] Fatehia.B.G et Mawia.A.H (2014) : 'Classification of breast tissue as normal or abnormal based on texture analysis of digital mammogram'. *Journal of Medical Imaging and Health Informatics*, 4 (5), 647-653.

[152] Jen.C et Yu.S (2015): 'Automatic detection of abnormal mammograms in mammographic images'. *Expert Systems with Applications*, 42(6):3048-3055.

[153] Kumar.S et Bandyopadhyay (2010): 'Detection of abnormal masses in mammogram images'. *International Journal of Computer Science and Information Technologies*.

[154] Sampaio.W.B, Diniz.E.M, Silva.A.C, Paiva.A.C et Gattass.M.d (2011): 'Detection of masses in mammogram images using CNN, geostatistic functions and SVM'. *Computers in Biology and Medicine*, page 653-664.

[155] www.mias.org

[156] I. Kanta Maitra, S. Nag S. K. Bandyopadhyay: 'Detection of Abnormal Masses using Divide and Conquer Algorithm in Digital Mammogram', *Int. J. Emerg. Sci*., 1(4), 767-786, December 2011.

[157] Chaza, C. : 'Fusion d'Informations par la Théorie de l'Évidence pour la Segmentation d'Images'. PhD. Dissertation, Université PARIS EST, 2016.

[158] M. Rombaut and Y. M. Zhu.: 'Study of Dempster-Shafer theory for image segmentation applications', *Image and vision computing*, vol. 20, pp. 15-23, 2002.

[159] M. ROMBAUT. : 'Fusion de données images segmentées à l'aide du formalisme de dempster shafer', in 17° Colloque sur le traitement du signal et des images, FRA, 1999, 1999.

[160] MIHOUBI Abdelhafid. : 'Classification Lithologique des Attributs Sismiques par les Réseaux de Neurones Artificiels', Mémoire de Magister, Université M'HAMED Bouguera Boumerdes, Le 26/04/2008.

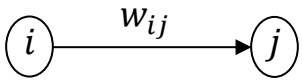
ANNEXES

Annex A: Error Correction Rules for Neural Networks

A.1. Hebb's rule

The first connectionist learning rule was unsupervised, it was inspired by biology. Indeed, in the late 1940s, Hebb proposed the idea that the brain adapts to its environment by modifying the efficiency of connections between neurons. This modification would be proportional to the simultaneous activity of the neurons related by the connection.

The hypothesis here is that a synapse improves its efficiency only when the activity of its two neurons is correlated. Hebb's principle remains biologically plausible today, and the modification of the synaptic weight was mathematically formulated as follows: [77]



$$w_{ij} = \eta . a_i a_j \qquad A.1$$

with:

- η (0< η <<1) is a constant.
- w_{ij} represents the synaptic weight between neurons i and j.
- a_i and a_j represent the correlation between the activation of neurons i and j of the network.

Applied to neuromimetic networks, this simple rule motivates neurons to correlate their activations with the values of the input patterns. One of the advantages of such behavior resides in the context of associative memories, and several connectionist models have been proposed with variants of the Heb's learning.

The activation correlation can have useful effects. However, the fact remains that it is a very limited learning criterion, since it does not take into account the nature of the task that it must carry out [77].

A.2. Delta's rule (or Widrow-Hoff's rule)

Its purpose is to make the network evolve towards the minimum of its error function (error made on all the examples). Learning is carried out by iteration (the weights are modified after each presented example), and we obtain the weight at the moment t+1 by the formula:

$$w(t+1) = w(t) + \eta \times (y^{des} - y) \cdot x \qquad A.2$$

Where: w is the weight, y^{des} is the theoretical output and y is the real output, x is the input and η is a learning coefficient (between 0 and 1) that can be reduced during the learning. This is actually a particular case of the gradient backpropagation algorithm [77].

A.3. Gradient's backpropagation

Most learning algorithms allow weights to be obtained by minimizing a differentiable cost function. The simplest method for minimizing is the gradient method.

To an input vector, we want to associate a desired output vector. If the weights have any values, the observed output vector is a priori different from that desired. We can associate with this difference the quadratic error. We can modify the network weights by minimizing this quadratic error.

Updating all weights of one layer requires the knowledge of the associated errors with each neuron of the next layer. We will apply the weight adjustment algorithm starting from the last layer (for which the errors are known) to the first, hence the name of this algorithm: Error gradient backpropagation algorithm [160]. Figure A.1 represents a simple network consisting of an input layer, a single hidden layer and an output layer.

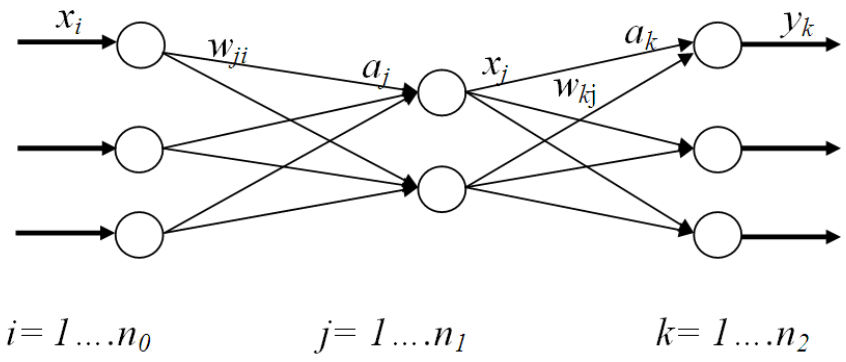


Figure A.1: A simple neural network.

- Cost function:

We present an example: X= [x_1, x_2, ..., x_{n_0}], and we calculate the corresponding output: Y= [y_1, ……, y_{n_2}].

The error vector will be: $e_k = y_k^{des} - y_k$ $A.3$

Where: y_k^{des} is the desired output.

Using the least squares calculation method, we calculate the operators w which minimize the sum of squares of the error function [160].

$$E = \frac{1}{2}\sum_{k=1}^{n_2} e_k^2 = \frac{1}{2}\sum_{k=1}^{n_2}(y_k^{des} - y_k)^2 \qquad A.4$$

The gradient of an error area of n dimensions is given by the vector defined by the partial derivatives of the error function. When a minimum is reached, the function: $\frac{\partial E}{\partial w}$ will be zero for all values of w.

Since the gradient is given by the partial derivatives, if we use as weights variation Δw proportional to the derivatives, we obtain the corresponding error: [160]

$$\Delta w = -\eta \frac{\partial E}{\partial w} \qquad A.5$$

With: $0 < \eta < 1$

We should not forget that we have hidden layers, we then back-propagate the committed error backwards until the input layer, so:

- For the output layer:

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial a_k} \cdot \frac{\partial a_k}{\partial w_{kj}} \qquad A.6$$

We have:

$$\frac{\partial E}{\partial y_k} = -\left(y_k^{des} - y_k\right), \qquad \text{because:} \quad E = \frac{1}{2}\sum_{k=1}^{n_2}(y_k^{des} - y_k)^2$$

$$\frac{\partial y_k}{\partial a_k} = f'(a_k), \qquad \text{because:} \quad y_k = f(a_k)$$

$$\frac{\partial a_k}{\partial w_{kj}} = x_j, \qquad \text{because:} \quad a_k = \sum_{j=1}^{n_1} w_{kj} \cdot x_j$$

We put: $$Errk = \frac{\partial E}{\partial a_k} = -\left(y_k^{des} - y_k\right) \cdot f'(a_k) \qquad A.7$$

Therefore: $$\frac{\partial E}{\partial W_{kj}} = Errk \cdot x_j \qquad A.8$$

- For the hidden layer :

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial y_k} \cdot \frac{\partial y_k}{\partial a_k} \cdot \frac{\partial a_k}{\partial x_j} \cdot \frac{\partial x_j}{\partial a_j} \cdot \frac{\partial a_j}{\partial w_{ji}} \qquad A.9$$

We have:

$$\frac{\partial a_k}{\partial x_j} = w_{kj} \quad \text{because:} \quad a_k = \sum_{j=1}^{n_1} w_{kj} \cdot x_j$$

$$\frac{\partial x_j}{\partial a_j} = f'(a_j) \quad \text{because:} \quad x_j = f(a_j)$$

$$\frac{\partial a_j}{\partial w_{ji}} = x_i \quad \text{because:} \quad a_j = \sum_{i=1}^{n_0} w_{ji} \cdot x_i$$

So:

$$\frac{\partial E}{\partial w_{ji}} = Errk \cdot w_{kj} \cdot f'(a_j) \cdot x_i$$

We put: $Errj = Errk \cdot w_{kj} \cdot f'(a_j)$ $\qquad A.10$

So:

$$\frac{\partial E}{\partial w_{ji}} = Errj \cdot x_i \qquad A.11$$

- Weight's change:

1- Output layer :

$$w_{kj} = w_{kj} + \Delta w_{kj} = w_{kj} - \eta \frac{\partial E}{\partial w_{kj}} = w_{kj} - \eta \cdot Errk \cdot x_j \qquad A.12$$

2- Hidden layer:

$$w_{ji} = w_{ji} - \eta \cdot Errj \cdot x_i \qquad A.13$$

Annex B: Belief Masses Modeling

There are methods of belief masses modeling that can be used specifically for example in image processing [157]:

***- Image's histogram**:*

This approach is interesting in the case where the histogram gives relevant information for the separation of classes. In the works of M. Rombaut and Y. Zhu [158,159], authors do an analyze of the image's histogram ***h***. The peaks of ***h*** are considered as simple hypotheses and the transitions between these hypotheses as multiple hypotheses.

Figure B.1 presents an example of histogram for a simple image. Figure B.2 presents two histograms from two different sources that will be fusion. And figure B.3 presents the simple and multiple hypotheses with the parameters used for the modeling of information concerning the element 'x'.

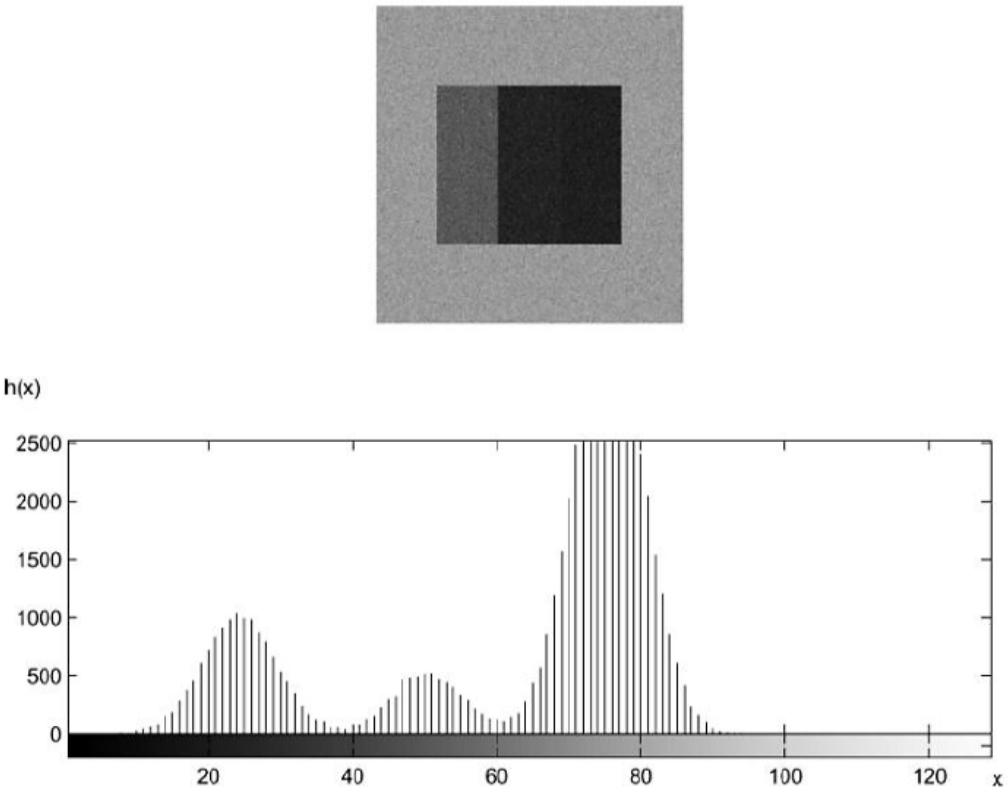


Figure B.1: Simple image and its histogram.

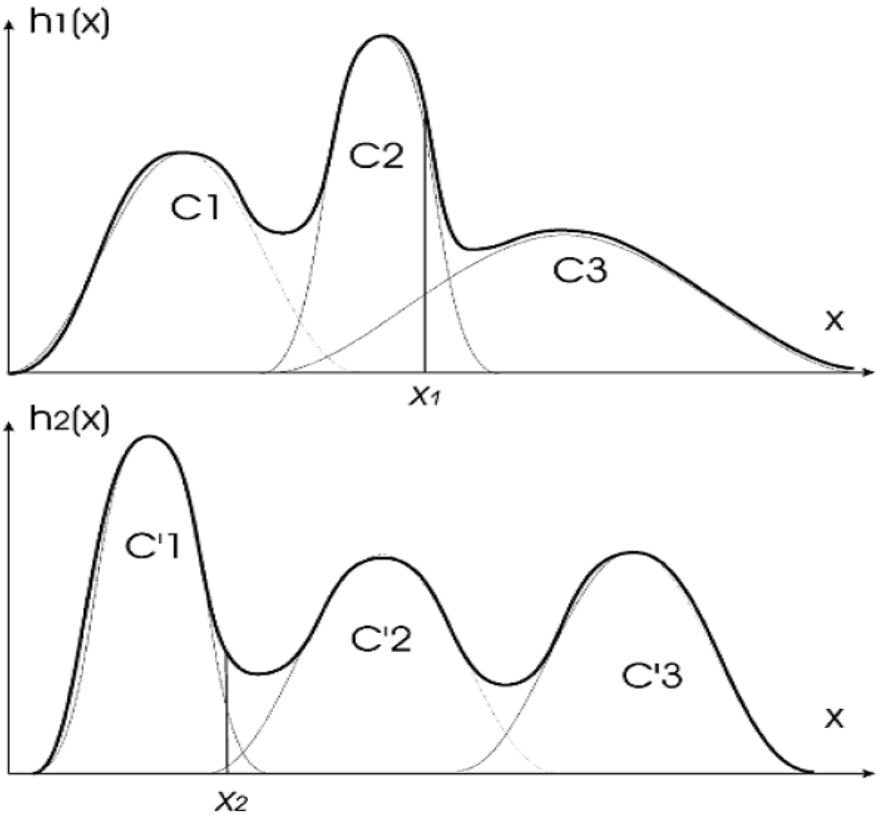


Figure B.2: Two examples of histogram modeled by the sum of Gaussian curves.

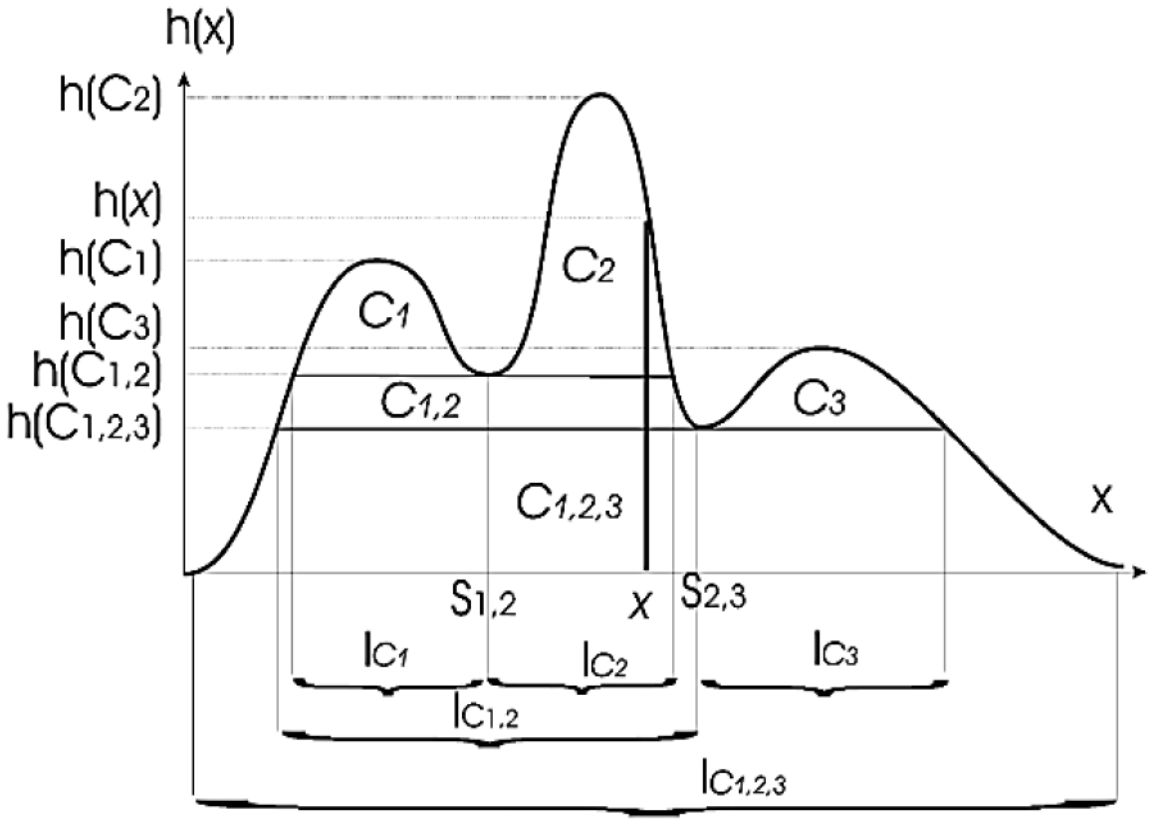


Figure B.3: Definition of parameters and method of masses' estimation.

From the example given in figure B.3, the new element 'x' falls in the intervals of classes C_2, $C_{1,2}$ and $C_{1,2,3}$. The modeling of masses for 'x' in this example is then given by the following expression:

$$m_{\mathrm{x}}(C_2) = K_{\mathrm{x}} \times \left[\, h(\mathrm{x}) - h\left(C_{1,2}\right)\right] \qquad B.1$$

$$m_{\mathrm{x}}\left(C_{1,2}\right) = K_{\mathrm{x}} \times \left[\, h\left(C_{1,2}\right) - h\left(C_{1,2,3}\right)\,\right] \qquad B.2$$

$$m_{\mathrm{x}}\left(C_{1,2,3}\right) = K_{\mathrm{x}} \times \left[\, h\left(C_{1,2,3}\right)\,\right] \qquad B.3$$

And in general case:

$$m_{\mathrm{x}}(H_1^{\mathrm{x}}) = K_{\mathrm{x}} \times [\, h(\mathrm{x}) - h(H_2^{\mathrm{x}})\,] \qquad B.4$$

...

$$m_{\mathrm{x}}(H_i^{\mathrm{x}}) = K_{\mathrm{x}} \times [\, h(H_i^{\mathrm{x}}) - h(H_{i+1}^{\mathrm{x}})\,] \qquad B.5$$

...

$$m_{\mathrm{x}}(H_n^{\mathrm{x}}) = K_{\mathrm{x}} \times [\, h(H_n^{\mathrm{x}})\,] \qquad B.6$$

With: $H_1^{\mathrm{x}} \subset H_2^{\mathrm{x}} \subset \cdots \subset H_n^{\mathrm{x}}$, and $K_{\mathrm{x}} = {}^{1}/_{h(\mathrm{x})}$: a standardization term.

In order to reduce the noise, often present in histograms, authors proposed to smooth the image before applying the method [157].

Annex C: Example of Evidence Theory Use

To implement the evidence theory and clarify its use, in this appendix we present a simple example on the application of the evidence theory and on the combination of information from different sources.

Suppose that three people, Paul (P), Jean (J) and Simone (S) are dining together in the house of their friend Antoine. After dinner they decided to listen to classical music in the living room. In this room Antoine keeps a very precious gem. He told his friends it was a gift from his grandmother. After a beautiful evening the three friends left and Antoine slept. The next day he discovers, shocked, that the gem has disappeared. No one can enter this room. One of the three friends took the gem.

The detectives and after interrogations allocated masses for the different elements of the discernment set.

The discernment set in this case is formed by the three friends:

$$\Omega = \{P, J, S\} \qquad C.1$$

Which implies that the power set is:

$$2^{\Omega} = \{\emptyset, P, J, S, \{P, J\}, \{P, S\}, \{J, S\}, \{P, J, S\}\} \qquad C.2$$

The masses assigned by the detectives are as follows:

Event	Subset	Masse
No one is guilty	$\emptyset$	0
P is guilty	P	0.2
J is guilty	J	0.1
S is guilty	S	0.1
P or *J* is guilty	$\{P, J\}$	0.2
P or *S* is guilty	$\{P, S\}$	0.2
S or *J* is guilty	$\{J, S\}$	0.1
One among the 3 is guilty	$\{P, J, S\}$	0.1

Table C.1 Masses assigned by the detective.

Credibility is the total belief that can be attributed to an element (see Equation III.9). For example, the credibility that Paul or John is guilty is expressed as follows:

$$bel(\{P,J\}) = m(P) + m(J) + m(\{P,J\}) = 0.2 + 0.1 + 0.2 = 0.5$$

Therefore:

Subset	Masse	Credibility (*bel*)
$\emptyset$	0	0
P	0.2	0.2
J	0.1	0.1
S	0.1	0.1
$\{P,J\}$	0.2	0.5
$\{P,S\}$	0.2	0.5
$\{J,S\}$	0.1	0.3
$\{P,J,S\}$	0.1	1

Table C.2 Credibility of the power set elements.

Plausibility is the maximum belief that can be attributed to an element (see Equation III.10). The plausibility that Paul or John is guilty is expressed as follows:

$$Pl(\{P,J\}) = m(P) + m(J) + m(\{P,J\}) + m(\{P,S\}) + m(\{J,S\}) + m(\{P,J,S\})$$
$$= 0.2 + 0.1 + 0.2 + 0.2 + 0.1 + 0.1 = 0.9$$

Therefore:

Subset	Masse	Plausibility (*Pl*)
$\emptyset$	0	0
P	0.2	0.7
J	0.1	0.5
S	0.1	0.5
$\{P,J\}$	0.2	0.9
$\{P,S\}$	0.2	0.9
$\{J,S\}$	0.1	0.8
$\{P,J,S\}$	0.1	1

Table C.3 Plausibility of the power set elements.

The probability that a set E is true falls in the interval *bel(E)* and *Pl(E)*. This interval measures the certainty that we have on a given element.

Subset	Credibility (*bel*)	Plausibility (*Pl*)
$\emptyset$	0	0
P	0.2	0.7
J	0.1	0.5
S	0.1	0.5
$\{P,J\}$	0.5	0.9
$\{P,S\}$	0.5	0.9
$\{J,S\}$	0.3	0.8
$\{P,J,S\}$	1	1

Table C.4 Credibility and Plausibility of the power set elements.

We assume now that there are two detectives and each one has an opinion concerning the situation. Each detective is considered as an independent source of information $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$.

Element	$\emptyset$	P	J	S	$\{P,J\}$	$\{P,S\}$	$\{J,S\}$	$\{P,J,S\}$
m_1	0	0.2	0.1	0.1	0.2	0.2	0.1	0.1
m_2	0	0.1	0.1	0.1	0.3	0.1	0.2	0.1

Table C.5 Masses assigned by the detectives.

Dempster's orthogonal combination law allows to fusion the information given by the two detectives in order to create a new global mass on the discernment set (See equation III.13). The normalization term k is equal to: (See equation III.14)

$$k = \sum_{B_1 \cap B_2 = \emptyset} m_1(B_1) \times m_2(B_2)$$

$$= m_1(P) \times m_2(J) + m_1(P) \times m_2(S) + m_1(P) \times m_2(\{J,S\})$$
$$+ m_1(J) \times m_2(P) + m_1(J) \times m_2(S) + m_1(J) \times m_2(\{P,S\})$$
$$+ m_1(S) \times m_2(P) + m_1(S) \times m_2(J) + m_1(S) \times m_2(\{P,J\})$$
$$+ m_1(\{P,J\}) \times m_2(S) + m_1(\{P,S\}) \times m_2(J) + m_1(\{J,S\}) \times m_2(P)$$

$$= 0.2 \times 0.1 + 0.2 \times 0.1 + 0.2 \times 0.2 + 0.1 \times 0.1 + 0.1 \times 0.1 + 0.1 \times 0.1$$
$$+ 0.1 \times 0.1 + 0.1 \times 0.1 + 0.1 \times 0.3 + 0.2 \times 0.1 + 0.2 \times 0.1 + 0.1 \times 0.1$$
$$= 0.21$$

Which implies that:

$$1/(1-k) = 1/(0.79)$$

The new mass assigned to the subset $\{P, J\}$ after the combination is calculated as follows:

$$m(\{P,J\}) = \frac{1}{1-k} \sum_{B_1 \cap B_2 = A} m_1(B_1) \times m_2(B_2)$$
$$= (1/0.79) \times [m_1(\{P,J\}) \times m_2(\{P,J\}) + m_1(\{P,J\}) \times m_2(\{P,J,S\})$$
$$+ m_1(\{P,J,S\}) \times m_2(\{P,J\})]$$
$$= (1/0.79) \times (0.2 \times 0.3 + 0.2 \times 0.1 + 0.1 \times 0.3)$$
$$= (0.11/0.79) = 0.14$$

The new masses assigned to the different subsets of the power set after the application of orthogonal law are then:

Element	$\emptyset$	P	J	S	$\{P,J\}$	$\{P,S\}$	$\{J,S\}$	$\{P,J,S\}$
m_1	0	0.2	0.1	0.1	0.2	0.2	0.1	0.1
m_2	0	0.1	0.1	0.1	0.3	0.1	0.2	0.1
m	0	0.316	0.227	0.177	0.139	0.063	0.063	0.013

Table C.6 Global masses of the power set elements.

Once the global masses are calculated, the global plausibility and credibility can be calculated like the previous cases and are equal to:

Element	$\emptyset$	P	J	S	$\{P,J\}$	$\{P,S\}$	$\{J,S\}$	$\{P,J,S\}$
m	0	0.316	0.227	0.177	0.139	0.063	0.063	0.013
bel	0	0.316	0.227	0.177	0.682	0.556	0.467	1
Pl	0	0.531	0.442	0.316	0.823	0.773	0.684	1

Table C.7 Global plausibility and credibility.

If we consider the maximum of plausibility as a decision law (see equation III.16) and in the case of simple hypotheses (P, J, S), the hypothesis P will be chosen as solution for this problem: $Pl(P) > Pl(J) > Pl(S)$. So, Paul is guilty with a plausibility = 0.531.

Annex D: Example of Fusion by Bayesian Approach

In order to clearly show all necessary steps to obtain a segmented fusion image, we will apply the Bayesian fusion algorithm on images obtained by simulation. These images, which are shown in figure D.1, are made up of a number of rectangles situated in the center of the image. There are three rectangles in each image. With the background, we are therefore in the presence of four objects, or four classes. The parameters of these classes are presented in Table D.1. The used noise is an additive Gaussian noise, of zero mean.

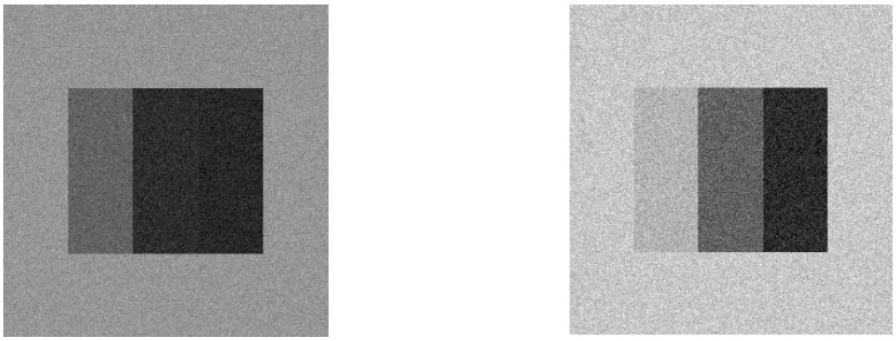


Figure D.1: Simulation images used to explain the fusion methodology using Bayes method.

	Class 1	*Class 2*	*Class 3*	*Class 4*	*Noise's Variance*
Population	70 %	10 %	10 %	10 %	
Mean (image 1)	45	50	100	150	10
Mean (image 2)	50	100	180	200	15

Table D.1: The used parameters during the numerical simulation.

To fusion the two images using the Bayesian approach, we take the gray level m_j of a pixel as primitive from the images. The different hypotheses here are the different regions of the simulated images.

The fusion process essentially contains four stages:

a) A first number of classes is defined;

b) Histogram models are constructed to allow the fusion of images;

c) Each pair of pixels in the two images is fusion using the Bayes rule.

d) A fusion image is obtained by taking the decision using the MAP criterion.

The parameters of classes are then adapted according to the current segmentation. Steps b) to d) are then repeated until the algorithm's convergence.

- Probability modeling

In absence of any information on classes distribution, we assume the hypothesis of the equiprobability of different classes. If we call k the number of classes supposed to be present in the image, we have that:

$$\forall i,\ P(H_i) = \frac{1}{k} \qquad D.1$$

Concerning the likelihood functions $P(m_j / H_i)$ which measure the probability of a considered gray level m_j given the hypothesis H_i, we place ourselves under the Gaussian hypothesis and we can use the following formula:

$$P(m_j / H_i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(- \frac{(m_j - \overline{H_i})^2}{2\sigma_i^2} \right) \qquad D.2$$

where $\overline{H_i}$ represents the mean and σ_i is the standard deviation of the Gaussian expression.

If we do not have other information on the likelihood function, we assume that, for a given image, each likelihood function has the same standard deviation σ, which is determined by the histogram's dynamics of each original image.

The histogram of each image is split into k intervals of the same length between the minimum gray level and the maximum gray level of the image. The means H_i of each class are the centers of these intervals. The standard deviation of the classes σ_i is calculated in a way that the intervals between the extremities of the considered class are equal to $2\sigma_i$, which represents 69% of the class's population.

Figure D.2 shows the initial histogram (represented by the thickest curve) of an image with three classes (represented by the three thinner curves). Note that the two images use the same initial histogram model.

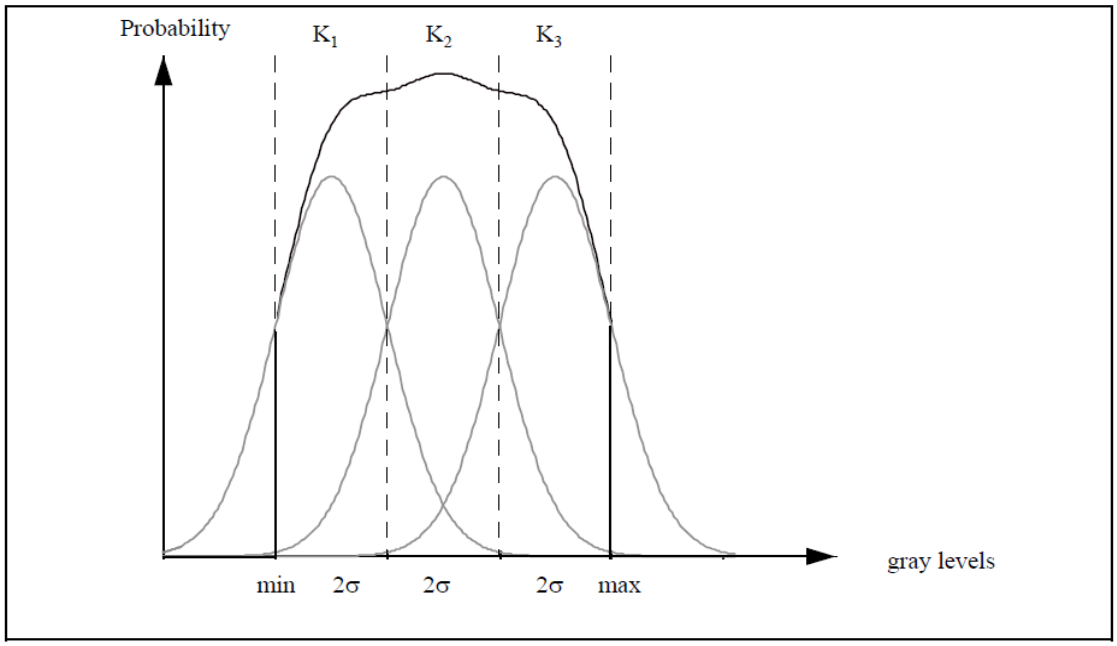


Figure D.2: Modeling of an initial histogram with three classes.

Each class is therefore determined by five parameters:

- Its population in percentage
- Its gray level average in the two original images
- Its variance in the two original images.

- *Decision making*

From the class modeling and histograms that we have defined above, we can apply the Bayes rule for all hypotheses and all grayscale couples (see equation III.33). We thus obtain k a posteriori probability, and we choose the hypothesis corresponding to the MAP.

After having obtained a first segmentation, we can then update the parameters of different classes ($\overline{H_i}$, σ_i and $P(H_i)$) from this classification. The parameters of each class are re-updated after each segmentation of the images. The process is reiterated until the convergence of the algorithm, that is to say until no pixel changes class from one iteration to another.

The fusion results are shown below. The first image shows the reference image, the second shows the result of our method and the third shows the error image.

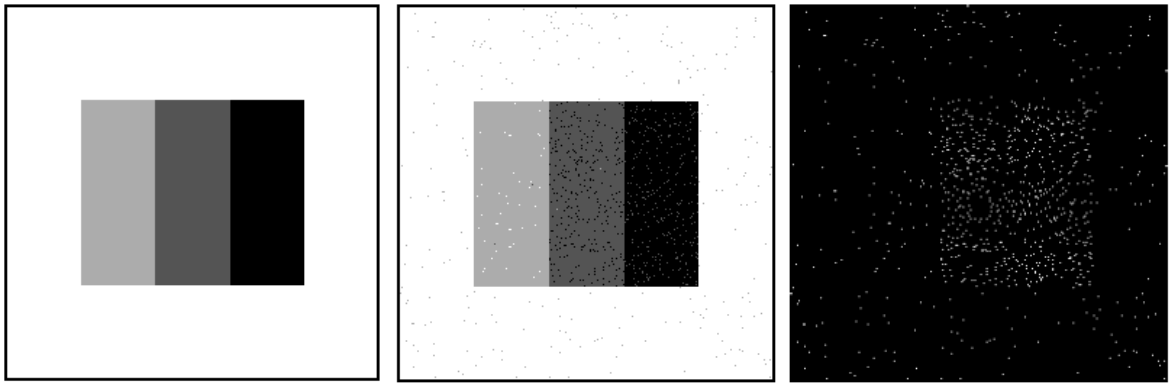


Figure D.3: Fusion results:
Left: reference image. Middle: fusion image. Right: error image.

For an image of 256 x 256 pixels, only 1.23% of the pixels were misclassified.