Multimedia Tools and Applications (2021) 80:24027–24052
https://doi.org/10.1007/s11042-021-10634-4

A new approach for texture segmentation based on the Gray Level Co-occurrence Matrix

Saliha Aouat[1] · Idir Ait-hammi[1] · Izem Hamouchene[1]

Received: 14 February 2020 / Revised: 18 December 2020 / Accepted: 4 February 2021 /
Published online: 30 March 2021

Abstract

Image processing is a very rich and important research area, which provides efficient solutions to many real and industrial problems. Texture analysis is one of the most interesting fields in image processing and pattern recognition. It became a very attractive research area these last years, especially after the growth and the advancement of technologies. This paper deals with texture analysis and unsupervised texture segmentation problem. The goal of this study is to develop a new segmentation method based on the textural features of the images. The proposed system is composed of different steps. First, the image is analyzed in each pixel using the Gray Level Co-occurrence Matrix (GLCM) feature extraction method. Four Haralick parameters (Haralick Proc IEEE 67(5):786–804, 1979) are calculated and represented in four matrix. After that, we applied the gradient to detect edges from the extracted images. In order to localize the area of the discontinuity of the texture, we proposed a new method for joining the edge and region growing. The proposed system is applied on several textured images and the obtained results are shown in the experimentation section. A number of experiments have been done with randomly generated textured images. The experiments have shown the efficiency of the proposed method compared to other existing methods and its robustness to enhance the segmentation precision of textured images.

Keywords Texture segmentation · GLCM · Haralick parameters · Fourier transform · Nagao filter · Gaussian filter · Median filter · Hysteresis thresholding

1 Introduction

Nowadays, the technology evolves very quickly. This has involved a big data which becomes difficult to process using traditional applications. Scientists and researchers began

✉ Izem Hamouchene
ihamouchene@usthb.dz

Saliha Aouat
saouat@usthb.dz

Idir Ait-hammi
iaithami@usthb.dz

[1] LRIA Laboratory, Computer Science Department, USTHB University, Algiers, Algeria

to develop advanced approaches to solve different problems automatically. One of the important problem which involved a great need of new advanced methods is image processing. This consists to analyze and recognize automatically different entities from an image based on its visual content automatically, without human intervention [16]. Thus, a lot of information can be processed in a short time. This can be very useful in industrial and medical fields such as printed documents, authentication, security field, satellite and medical imagery applications. Image processing can be sub-divided into three categories: shape, color and texture. In this study, we are interested on the textural features of the image.

Texture analysis is a fundamental field in image processing because most of real life objects and their surfaces are textured in nature. The goal of the texture analysis is to describe the texture by a mathematical model, which allows identifying different textures and objects in the image. The first studies were proposed by Harlick [16]. The proposed texture analysis methods might be categorized into four kinds of approaches: Structural approaches, Statistical approaches, Model approaches and Transformed approaches [34]. Image segmentation plays a key role in image processing and computer vision. It is performed before the analysis and decision steps. Hence, the need to use texture features in order to improve the segmentation of images which contain natural scenes or textured areas.

This this work, we proposed a noval image segmentation method. The proposed method follows a series of steps. First, the application of the Gray Level Co-occurrence Matrix (GLCM), which is a statistical approach, is studied and analyzed using different geometric transformations. Four parameters are calculated and represented in four images. After that, a segmentation method of derivative type, which is based on Euclidian distance, is applied to detect each part of the image which contains different textural features. In the last step, we proposed a new process in order to improvement of the region growing segmentation. First, we applied a smoothing filter to remove noise. After that, we used a Hysteresis thresholding unlike a static threshold to make our system more robust. Then, we applied a morphological closing operator in order to enlarge the boundaries of the bright regions in the image. After that, to increase the precision of our system, we proposed a method to join the edge of the segmentation. Finally, we proposed a growing region method in order to segment differents regions of the texture. To summarize, The extracted Haralick measures from the GLCM are improved using : several filters, a hysteresis thresholding instead of a simple thresholding, the discontinuities edges are reduced using morphological operations, the proposed joining edge method and growing region increases precision of the segmentation.

The paper is organized as follows: The next section presents an overview of texture analysis methods. Section 3 illustrates an overview of different image segmentation methods. In Section 4, we present and explain our proposed edge detection and texture segmentation methods. Section 5 illustrates some obtained experimental results. Finally, section six summarizes our contribution; discusses the strengths and the weaknesses of our segmentation method, illustrates some perspectives that can be added, and concludes this paper.

2 Texture analysis : Overview

In the world, the objects that surround us (which are natural or artificial) are covered with textured surfaces. These surfaces are often described as micro, macro, regular, periodic, stochastic and random texture [32]. These features allow us to better identify areas of an image and understand our environment. Although, there is not a strict definition of the texture in the literature, it is easily perceived by humans and extracts a lot of visual information.

Initially the texture has been defined as a field of the image, which appears as a coherent and homogeneous area. Other defined it as the spatial variation of a gray level in an area of the image. Thus, a set of connected pixels having a gray level which is repeated in an area of the image constitutes a textured area [30]. Also to analyse the texture a local and global features should can be applied [6]. Recent and novel texture analysis methods have been proposed and applied in several domains such as satellite, medical imagery and biometrics [7, 13].

Haralick extended the definition by describing the texture as a phenomenon which has two dimensions. The first is the description of a basic primitive (pattern) from which texture is formed. The second dimension is the description of the spatial organization of these primitives [16]. Three main categories of the texture can be denoted: structural, stochastic and directional textures.

- The structural textures (or regular) are characterized by their deterministic nature. It considers a spatial distribution of basic element (called Texton [19]), and repeated in different directions in space according to a certain rules.
- Stochastic textures look like noise and have an anarchic appearance but remain globally homogeneous. This is due to the randomly scattered, over the image, of the primitives. So, we cannot extract the repeated basic element on the image. This category has evolved a new research works based on statistical analysis methods. In this case, the texture is considered as a two dimensional stochastic process.
- Directional textures are not totally random and not characterized by a structural element. They are characterized by a specific orientation.

Almost all textures of our real world are irregular. The stochastic definition of the texture is often the most appropriate. For this reason, our research is based on statistical approaches.

A large overview of texture analysis methods have been made in [8]. Authors split their works in different order measures and discuss the advantage and weakness of each order. They explain that the most commonly used methods to analyze the texture are based on statistical measures. The statistical approaches assume that the texture is a stochastic process. The goal is to extract statistical measures to describe the texture. Thus, the texture is defined by the distribution of its gray level pixels. The order of the measure depends on the number of pixels used in the statistical analyze. If one pixel is used, it is a first order measure. Thus, if two pixels are used in the analysis process, it is a second order measure.

2.1 First order measure

The first order statistics measure is based on the gray level and intensity of the individual pixel values. This measure is extracted from the intensities histogram. Thus, the textured image can be seen as a function f(x,y). This function takes discrete values i=0,1,..,G where G is the total number of the intensity levels in the textured image. The intensity-level histogram is constructed by the occurrence number of each gray level, following this formula.

$$h(i) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} \varphi(f(x, y), i) \tag{1}$$

$$\varphi(a, b) = \begin{cases} 1, & a = b \\ 0, & a \neq b \end{cases} \tag{2}$$

To obtain the occurrence probability of the intensity levels, h(i) is divided by the total number of pixels. The shape of the histogram can indicate the low contrast of the image.

Different parameters can be extracted from the histogram such as the central moment [24, 28].

The first order statistics measure, which is based on the distribution of gray levels of the texture, has an important disadvantage. This order does not take into account the spatial distribution of gray levels. Indeed, we can find two different textures having the same gray level distribution as in Fig. 1.

To resolve this problem, second order statistical methods were proposed. These methods are based on the geometry of the gray level arrangements and the spatial distribution of gray levels.

2.2 Second order measure

The first order measure, which is based only on one pixel, is not enough to describe the texture. Thus, it is necessary to use higher order which considers more pixels to improve the analysis. Second order measure deals with a pair of pixels in the image. The relationship between pixels allows calculating the statistical attributes of the texture. The second order measure is the most order used due to its efficiency and its similarity to the human perception.

The most popular statistical method of the second order measure is the Gray-Level Co-occurrence Matrices (GLCM) [8, 16]. The GLCM has been used in various and recent works [15, 23, 33]. GLCM has obtained better results compared to efficient method like wavelet packets on texture classification [35]. The idea of this method is to calculate the number of apparition of two pixels (P_i, P_j) separated by a distance d and characterized by a direction θ. The matrix C_d, θ(i,j) estimates of the joint probability of each two pixels. This matrix is calculated as follow (Fig. 2).

$$C_{\Delta x,\Delta y}(i,y)=\sum_{p=1}^{M}\sum_{q=1}^{N}\begin{cases}1, & if I(p,q)=i\\ 0, & otherwise\end{cases} \tag{3}$$

According to this Formula, the parameters d and θ of the Co-occurrence matrix are very important and affect greatly the quality of the results. Therefore, Haralik [16] suggested to use the distance d=1,2 and the orientations 0°, 45°, 90° and 135°. Figure 3 shows how the GLCM works.

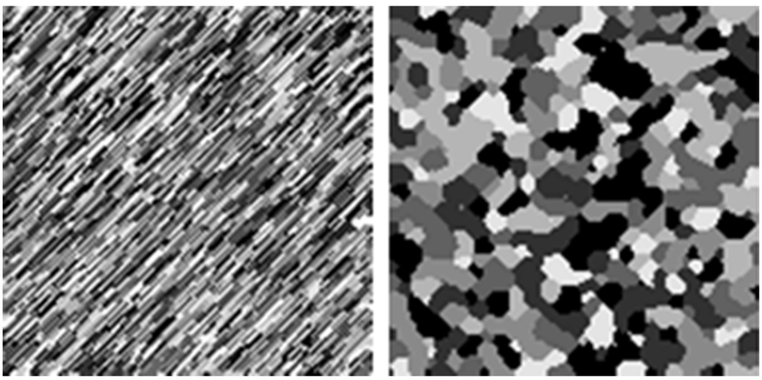


Fig. 1 Two textures having the same gray level distribution but visually different

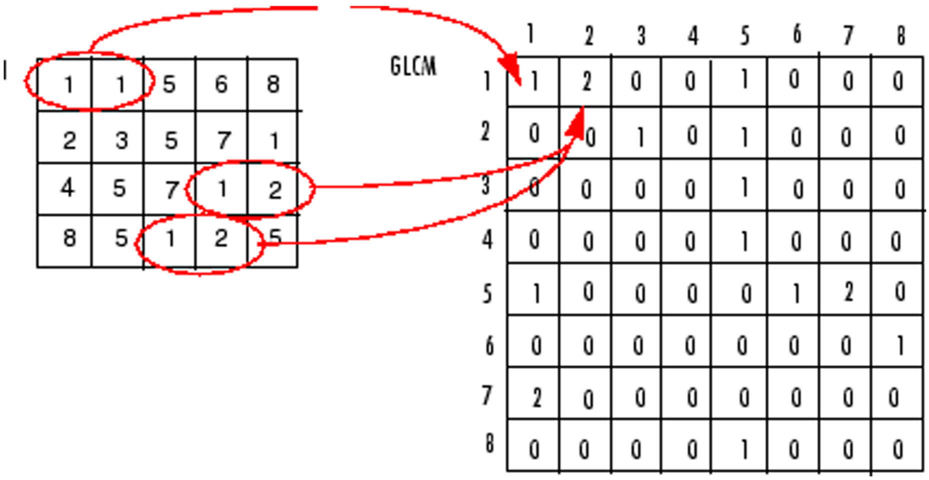


Fig. 2 Calculation of the GLCM matrix from a grayscale image [16] (distance = 1, orientations = 0°)

For exemple, we have the gray level 1 followed by value 2 two times in our image (shown with red circle in the Fig. 3). So, the the GLCM matrix (ligne 1, column 2) we put 2 (number of occurence of value 1 followed by value 2 in our image). Thus, the first value represent the number of ligne in the GLCM matrix and the second value represent the column.

Too much information can be extracted from the GLCM than cannot be analyzed in a short time. Thus, Haralick [16] proposed thirteen texture features from GLCM for an image

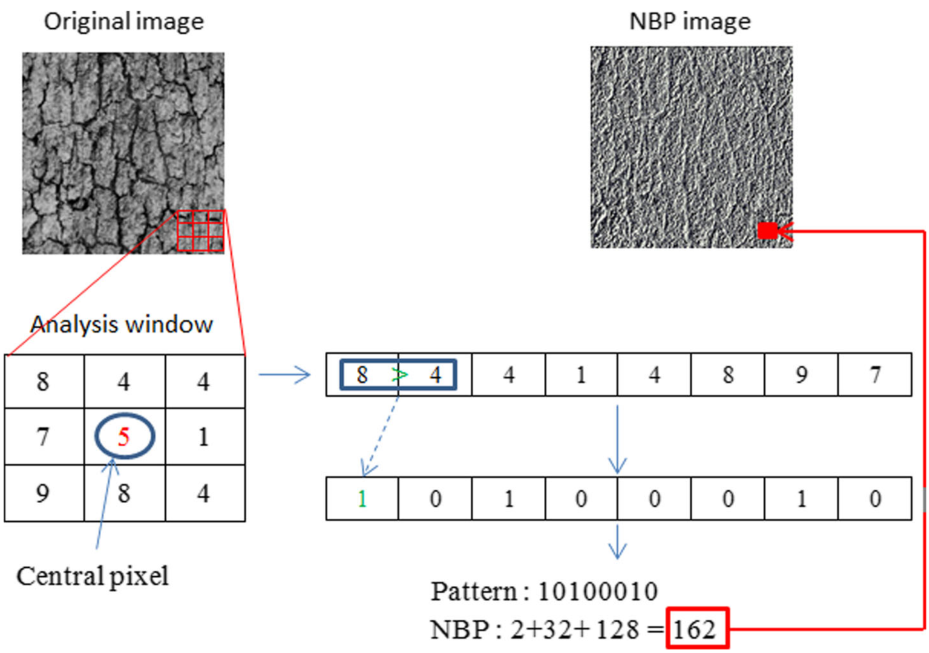


Fig. 3 Extraction of the NBP code of one central pixel

to describe the texture. These features reduce the size of the information contained in the GLCM (and allow better description and distinction between textures). The most commonly used features are Energy, Contrast, Inverse Different Moment and Correlation.

Energy: Energy is also known as Angular Second Moment (ASM) feature. It measures the uniformity of the textured image. Thus, when pixels are very similar, the energy value will be high (eg. uniform image or periodic texture).

$$E = \sum_i \sum_j C_{d,\theta}(i, j)^2 \tag{4}$$

Contrast: The contrast (or inertia) measures local variations in gray levels. It is a measure of intensity variation between a pixel and its neighbor. If these variations are important, the value of the contrast will be high. This parameter also enables to characterize the value dispersion of the GLCM compared to its main diagonal.

$$Con = \sum_i \sum_j (i - j)^2 C_{d,\theta}(i, j) \tag{5}$$

Inverse Different Moment (IDM): IDM is also called homogeneity (HOM), it measures the local homogeneity of an image. IDM feature obtains the measures of the closeness of the distribution of the GLCM elements compared to the GLCM main diagonal. This parameter has an opposite behavior of contrast. When texture is more homogeneous, this parameter will be high.

$$Hom = \sum_i \sum_j \frac{1}{1 + (i - j)^2} * C_{d,\theta}(i, j) \tag{6}$$

Correlation: Correlation feature shows the linear dependency (if some columns of lines of the matrix are equal) of gray level values in the GLCM. Indeed, higher the values are uniformly distributed in the GLCM, most the correlation is important.

$$Cor = \frac{\sum_i \sum_j I(i, j) * C_{d,\theta}(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{7}$$

Where μ_x, μ_y, σ_x, σ_y denote the mean and the standard deviations of the row and column sums respectively.

A set of a GLCM can be produced depending on the parameters (d,θ). Thus, a set of parameters are applied and a GLCM is calculated according to each chosen parameter to describe better the texture. However, the number of the required values of (d,θ) and the adequate values still remains an unresolved problem. This is the main weakness of this method. Experimental studies of this paper determine the most relevant values of the GLCM, which get the relevant local and global features to describe better the texture.

Since texture is based on the relation between a pixel and its neighbors, [5, 12] proposed a noval method (NBP) based on neighbors of each pixel. They also proposed a rotation invariant version (RINBP method) [9, 11, 13]. First, a 3x3 analyses window is used. Each neighbor of the central pixel is thresholded by the value of the next neighbor. If the value of the neighbor is greater than the next ones, this neighbor is encoded by 1. Otherwise, the neighbor is encoded by 0. The obtained binary code (multiplied by the weights matrix in order to be converted into a decimal number) is the value of the new central pixel in the output image. This process is illustrated in the following figure.

Finally, an RINBP histogram of the output image is calculated to describe the texture [10].

3 Texture segmentation : overview

Segmentation subdivides the image into homogeneous regions. This makes the description of the information easier to use. Formally, the segmentation is the action of subdividing an image into sets of pixels called regions (R_i) which constitutes the same texture [29]. Each region satisfies the following properties (Fig. 4)

– Can't have an empty region (each region must contains at least one pixel)

$$\forall_i,\, R_i \neq \emptyset \tag{8}$$

– Each pixel is associated to only one region

$$\forall_{i,j},\, R_i \cap R_j = \emptyset \tag{9}$$

– The union of all regions represent the whole image

$$\bigcup_{i=0}^{n} R_i = 1 \tag{10}$$

3.1 Region based segmentation methodes

Region based segmentation is also called "Similarity Based Segmentation" [3]. The idea is to regroup all pixels related to an object [20] and this area should be closed. Each pixel is related to one region and is taken into consideration [17]. Algorithm based on region was introduced by Zucker in 1976 [39]. The first step is to select a set of pixels from the image that have similar criteria such as grey level intensity. After that, enlarging these regions by appending each neighboring pixels that have similar properties to the first pixels. Finally, stop the region growing when no more pixels met the criteria for inclusion in that region (as shown in Fig. 5).

In Fig. 5, we can notice the starting pixel with the red color in the first image. The second and the third images are successive stages in the growth region. The last image is the obtained segmentation result using this method.

In [14, 15] authors proposed a dynamic decomposition method. First, all the image is considered as a main window. After that, for each iteration, the size f the main window is redused and all possible windows having the same size of MW are generated. For each generated window, a feature description method (NBP) is appyed to describe the texture. Finnaly, all relevent window are saved.

Active contour method is used for region based semgentation. In [37] authors proposed region-based segmentation using active contour and local patch similarity measure. First, they construct a local patch similarity measure as the spatial constraint. After that, they

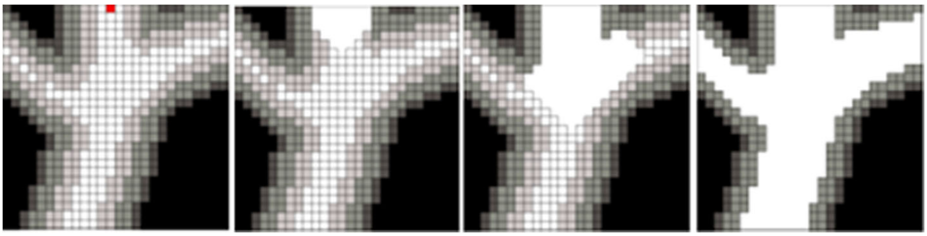


Fig. 4 Exemple of region growing segmentation method (started from the red point)

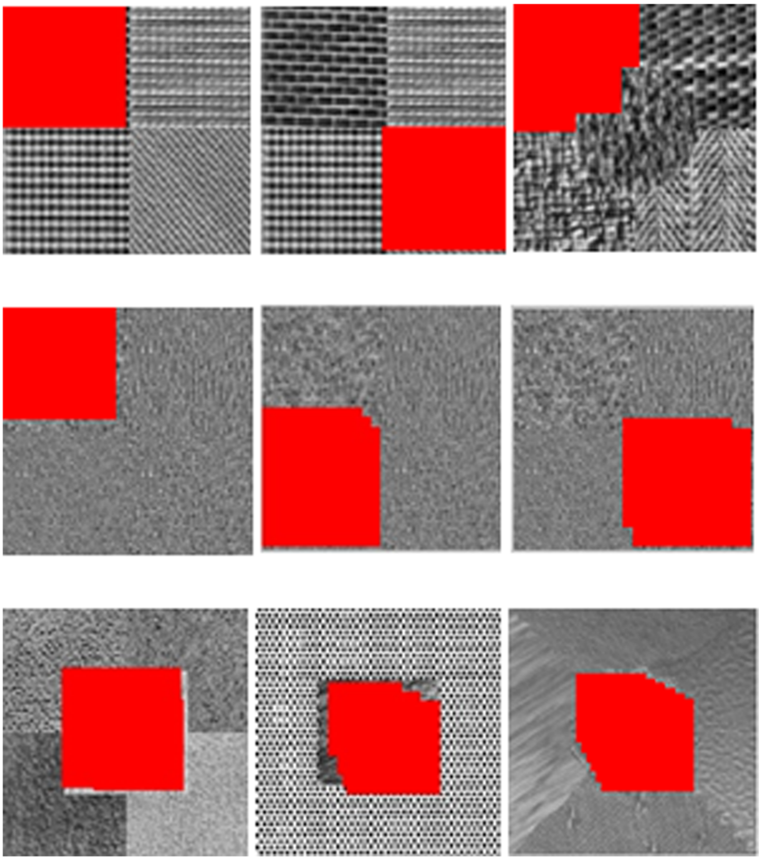


Fig. 5 Texture retrieval using dynamic decomposition segmentation

construct the proposed model by integrating the patch similarity measure into a region-based active contour model. Finally, they add a regularization information term to the objective function on order to guarentee the smoothness of the curve evolution.

3.2 Edge based segmentation methods

Edge based segmentation methods are characterized by the fact that they take into account only the information of the boundaries. Thus, these methods attempt to detect the edges between different regions, which have rapid transitions in intensity and extracted [31] by identifying the discontinuities in the image. After that, the detected pixels are linked to form closed object boundaries. Various edge detectors methods have been developed and used to segment the image [21].

Derivative filters are based on the study of local gray level variations. Based on the classical definition of the one-dimensional derivative of function, the derivation of the image function allows detecting these local variations. The most known approaches use the Gradient, Laplacian, deformable models, etc. In modern digital image processing one of the most common used techniques is the Gradient filter. The gradient of an image is the partial derivative of the gray level function *f*. Let it be the Image function f (x, y) defined in a two-dimensional space; we can define partial derivatives according to definition variables of *f*.

$$\frac{\partial f(x, y)}{\delta x} = \lim_{h_x \to 0} \frac{f(x + h_x, y) - f(x, y)}{h_x} \tag{11}$$

$$\frac{\partial f(x, y)}{\delta y} = \lim_{h_y \to 0} \frac{f(x, y + h_y) - f(x, y)}{h_y} \tag{12}$$

In order to perform edge detection, the gradients of the image in two orthogonal directions are calculated. After that, select the strongest detected edges. So, the pixels having the higher contrast are selected to segment the regions on the image.

In order to calculate the magnitude of the gradient a convolution of two masks is applied on the two orthogonal direction images. Thus, horizontal and vertical edges are detected.

We can find in [1] a comparative study of six methods for classification of texture 3D. Authors selected representative state of the art methods and used them on 572 synthetic textured mesh models. In [22] autors present a guide for developing a robust methods in order to extracts key geometric and topological information from signals and images.

Active contour model (ACM) is also widly used for edge detection [38]. Due to due to low contrast and complex noises in medical images, authors [38] proposed an edge-based active contour model segmentation. They also integrated an aptive perturbation into their framework of the edge-based ACM.

4 Proposed method

In this section, we will explian our proposed method and its different parameters. Some examples and illustrations will be given. Our approach is based on the GLCM method and the extracted Haralick features. First, we will study the behavior of the GLCM against translation and rotation. Based on this study we will adapt and take into consideration the most relevant parameters of the GLCM.

4.1 Study of the co-occurrence matrix

In this part, we will study the ability of the co-occurrence matrix to characterize the texture after applying the following geometric transformations on the image: change of scale and rotation. To do this, we chose one textured image. After that, the Haralick features are extracted. Finally, the Euclidian distance is calculated between the original image and the transformed image.

Rotation: The textured images have been rotated using a set of angles from 0° to 360° in 5° increments. After that, the obtained Euclidian distances are calculated. Figure 6 illustrates the distance for each angle.

Figure 6 illustrates the weakness of the GLCM method against the rotation. The axes ox represents the angles and the axes oy represent the dissimilarity measures. We can notice that some angles have the distance value close to 0 such as 90°, 180° and 270°.

Scale change: Scaling is the process used to zoom or reduce the size of an image. The obtained results are illustrated in Fig. 7. The ox axes represent the zoom value and the oy axes the dissimilarity value.

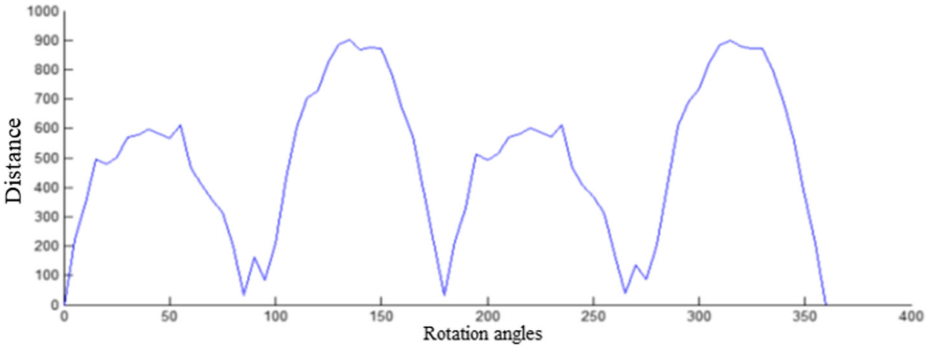


Fig. 6 Euclidian distance between the original image and rotated images

Figure 7 have shown that the GLCM is not robust against scale translation. Thus, each scale, the values of the GLCM matrix will changes depending on the number of pixels and the occurrence of the relation between them.

After studying the ability of the GLCM method and its weakness against some geometric transformations, we will explain our proposed segmentation method and the improvement made to solve this problem. Our system is based on two main phases : Segmentation by edge detection and region growing.

4.2 Segmentation by edge detection

In order to segment the image by edge, we used the GLCM method to characterize each pixel of the image. We proposed a process composed of four steps to detect the edges.

4.2.1 Step 1: gray-level reduction

The goal of this step is to reduce the gray level of the image without loss of the textural information. This is for the purpose of reducing the computation time of the GLCM. Reducing the gray levels involves the reducing of the GLCM size. If we have 256 gray levels, we will calculate a GLCM matrix of 256x256. Thus, we do our test by reducing the gray level

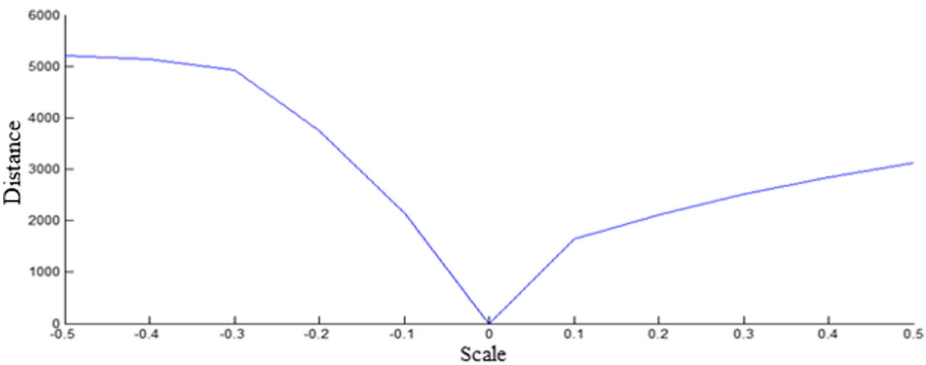


Fig. 7 Scale transformation

of the input image by 32, 16 and 8 gray levels. In our experiments we noticed that the best parameter is 16 gray levels.

4.2.2 Step 2: calculate the GLCM

In this step we cover the image with an analysis window. The size of the analysis window is T and the offset is d. The goal is to calculate the GLCM of each analysis window and extract the Haralick features. These features will be assigned to the central point of the analysis window. In our study we use four Haralick features which are the Energy, homogeneity, Correlation and Contrast. Finally, we obtain four output images where each one represents one of the Haralick feature. One example is illustrated in Fig. 8.

One key problem of this technique is the size of the analysis window. Indeed, if the analysis window's size is too big, one block may contain more than one texture. In the other case, if the size of the analysis window is too small, the researched texture could not be recognized. So, what is the best size of the analysis window?

In order to find the best size of the analysis window that characterizes efficiently the texture, we have done a series of tests with different windows' sizes (5x5, 9x9, 15x15, 20x20). After that, we display the obtained images of each Haralick feature (a- Energy, b- Homogeneous, c- Correlation and d- Contrast). The resulted images are illustrated in 3D to illustrate better different textures.

We can notice (Fig. 9) that for the sizes of 5x5 and 9x9 the GLCM does not recognize well the texture and contains a border detection problem. But, when the size of the analysis window is 15x15 and 20x20 (Fig. 10), the different textures are well recognized and separated. We can notice also that the features of the same texture converge to the same value

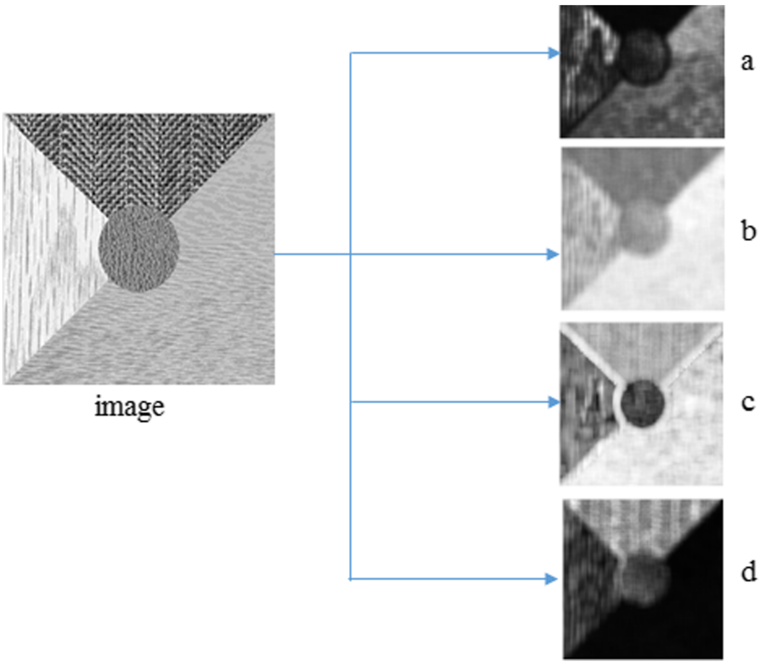


Fig. 8 Haralick measures. **a** Energy, **b** Homogeneous, **c** Correlation, **d** Contrast

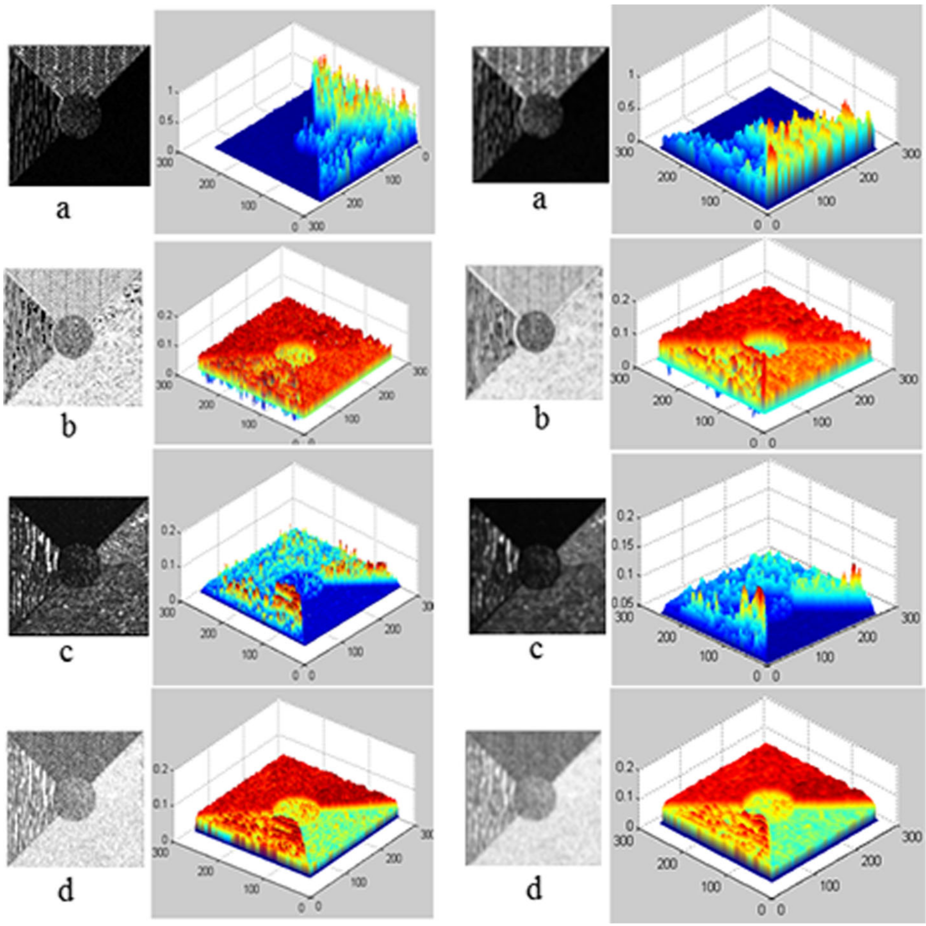


Fig. 9 3D representation of GLCM mareix (analysis window's size 5x5 in the left and 9x9 in the right)

when the size is high (15x15 and 20x20 pixels) but the features of the same texture are very different when the size of the analysis window is small. We also used a bigger analysis window, but the textures are not well recognised. This because one window contains more than one texture. The size of the window depends mainly if the original images contain several textures in a small area or not.

Based on this result, we noticed that the 15x15 size gives a better and homogeneous result than 20x20. So, we fixed the size of the analysis window to 15x15 in our approach.

We have fixed the first problem which is the size of the analysis window, a second parameter has to be fixed too. What is the best offset to cover the textured image (second parameter of the GLCM)?

To analyse this problem, we generate one textured image. This image contains two textures repeated periodically and reduced width. After that, we calculate the Energy features from the extracted GLCM. The obtained results are illustrated in Fig. 11.

We can notice from Fig. 11 that the distances from 1 to 5 do not affect greatly the detection of the different regions. But, more the offset is increased, the resulting image quality

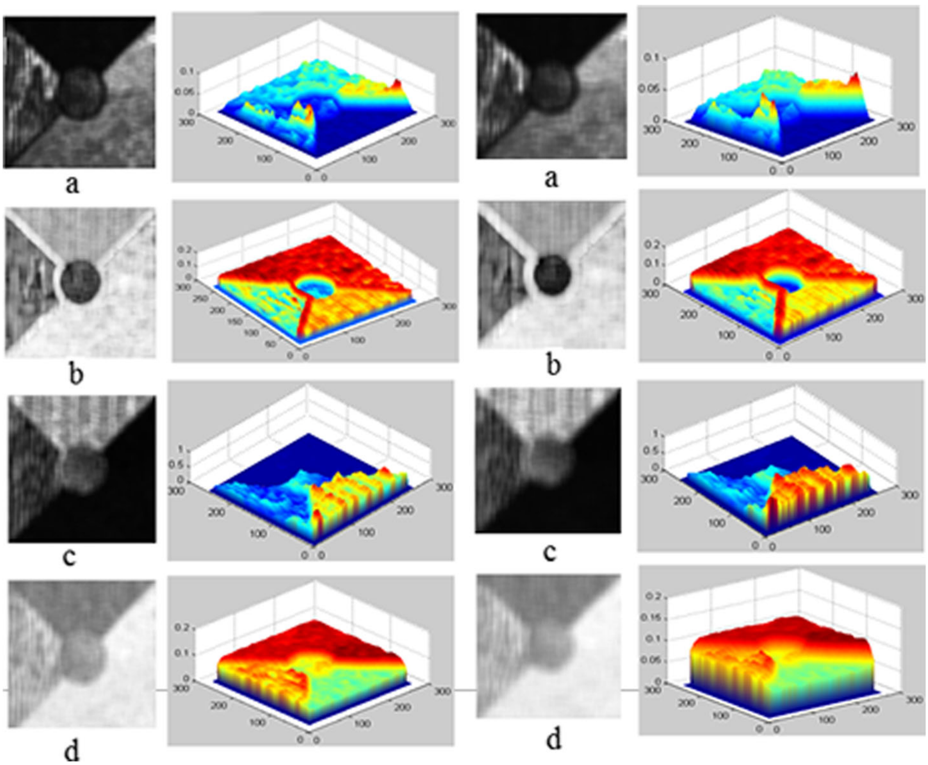


Fig. 10 3D representation of GLCM mareix (analysis window's size 15x15 in the left and 20x20 in the right)

becomes bad. This will affect the precision of the segmentation. The offset 1, 2 and 3 give good and almost similar results. In our approach, we maintain the offset 3 in order to reduce the computation time.

4.2.3 Step 3: edge detection

In this step, we use the gradient to detect the edges from the four extracted images (Energy, homogeneity, Correlation and Contrast). After that, two images are calculated (Vertical and Horizontal direction). Finally, we calculate the mean of the two resulted images and combine them into one output image. This process is illustrated in Fig. 12 (As exemple, applied in one haralick feature (Energy)).

Figure 12a and b are the Horizontal and Vertical difference of the Energy feature extracted from the GLCM. After extractiong the gradient image, we calculate the difference of a pixel with its vertical and horizontal neighbors (left neighbor for the horizontal, below neighbor for vertical). The obtained image illustrate the first result of edge detection.

4.2.4 Step 4: apply threshold

This step aims to increase the detect edges of different regions. To do this, we applied a threshold S on the obtained image from Step 3. If the value of the pixel is higher than S, this pixel is replaced by 255. In the other case, if the pixel is lower than S, this pixel is replaced

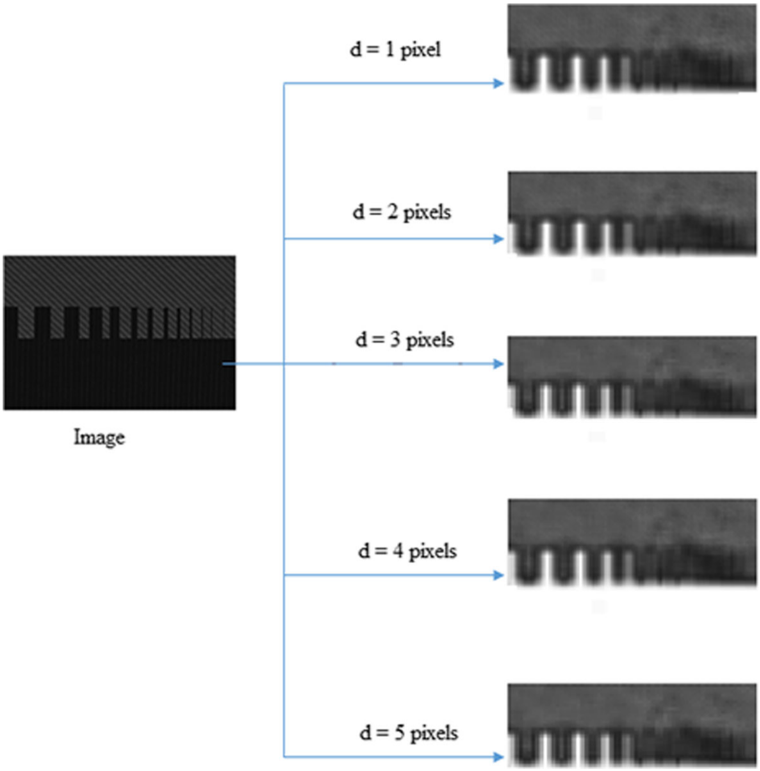


Fig. 11 Energy feature of the GLCM matrix using different offset

by the value 0. The value of the threshold S is important and affects greatly the results. However, the choice of this threshold in texture analysis is still a difficult problem to resolve. There are several proposed methods to fix the threshold automatically such as Otsu method [27]. In [12] autors proposed a noval method based on the relevance matrix to solve the fixed threshold value. Each analysis iteration increase the relevance of the searched texture. In our work, we tested different values of the threshold from 0 to 255 to detect the best threshold. Different obtained results are illustrated in Fig. 13 (using the Energy features).

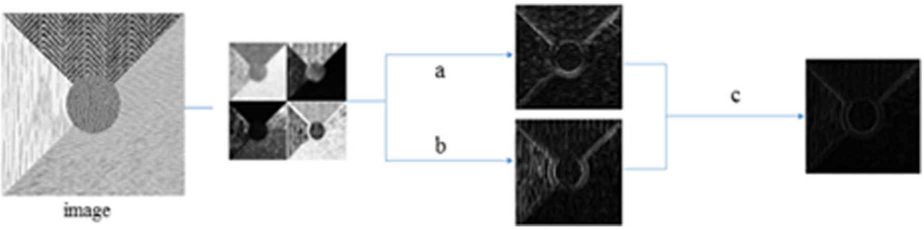


Fig. 12 Gradient image of the energy GLCM feature **a** Vertical difference of the Gradient's energy **b** Horizontal difference of the Gradient's energy **c** Mean of the vertical and horizontal gradient

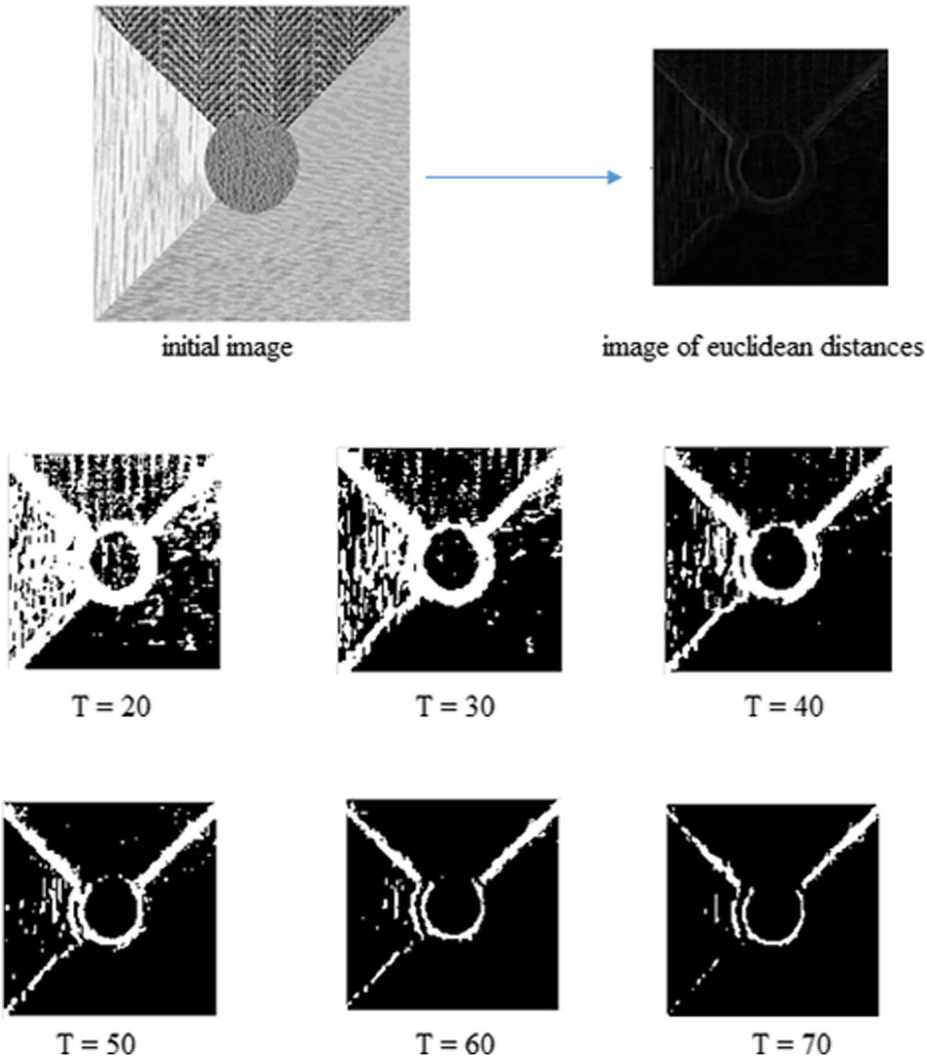


Fig. 13 Obtained results using different thresholds (20, 30, 40, 50, 60 and 70) on the obtained image from step 3

We can notice from Fig. 13 that the best threshold in our study is equal to 50. In nex steps, we will improve the threshold by proposing a new approche to apply dynamically the threshold.

4.3 Improvement of the region growing segmentation

In order to improve the region growing method, we proposed to apply a series of processes on the four transformed images (Fig. 8). All following process in this section is therefore applied on the energy, the homogeneous, the correlation, and the contrast obtained from the step 3 (Fig. 12).

Application of filters In order to improve the segmentation method, smoothing filters are applied to remove noise. These filters are applied on the four generated images (Energy, homogeneity, Correlation and Contrast). Mean, Gaussian, median and nagao filters are applied. Nagao filter is used to smooth the image and conserve the edges. Nagao filter replaces each pixel by the mean of particular windows [26]. There are nine predefined Nagao windows. Each window contains 9 pixels (central pixel and 8 neighbors pixels). This involves choosing the most suitable window from among a number of predefined windows. Each configuration gives diffetent results depending of which one is used.

Hysteresis thresholding Hysteresis thresholding [36], unlike the standard thresholding method, is not equal at all point of the image. The goal is to keep the strongest edges of the image and keep their continuity. First, two thresholds are used, high threshold S_h and low threshold S_l. The high threshold is used to detect the strongest edges of the image. On the other hand, the low threshold allows highlighting the weak edges of the image. Each weak edge is conserved only if its neighbor is the strong edge detected by S_h.

Algorithm 1 Hysteresis thresholding algorithm [37]

```
for each pixel do
    if edge > S_h then
        edge = 255
    else
        if edge > S_b and edge < S_h then
            edge = 128
        else
            if edge < S_h then
                edge = 0
            end if
        end if
    end if
end for
for each pixel do
    if edge = 128 and neighbor(edge) = 255 then
        edge = 255
    else
        edge = 0
    end if
end for
```

Figure 14 illustrates the obtained results between the standard thresholding and the Hysteresis thresholding algorithms.

After finding the edges of the image, edges are not necessarily closed. To resolve this problem we applied a morphological closing.

Morphological closing Morphological Closing is an important operator from the field of mathematical morphology. It can be derived from the fundamental operations of erosion and dilation and applied to binary images. Closing tends to enlarge the boundaries of the bright regions in the image. It is defined simply as dilation followed by erosion using the same structuring element for both operations. The structuring element used is generally a

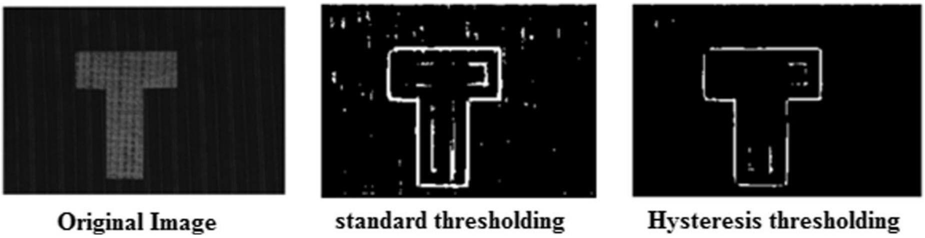


Fig. 14 Improvement of the edge detection using the Hysteresis thresholding algorithm compared to standard

window with size of 3x3 pixels. Figure 15 illustrates the application of the closing method on an image.

Even after applying the closing method on the image, the edges are not totally recognized. The closing process is a difficult problem to solve. Some fast detected edges affect the segmentation result. To solve this problem, we proposed a pre-processing steps are applied in order to improve the segmentation. We proposed a method to join edges.

Joining the edge points We can notice from the previously obtained results that there are some not closed edges. This is due to a small discontinuity of the edge. In order to improve this weakness, we proposed a novel method to solve this problem of discontinuity. We propose to join the nearest points according to a distance d using a circle analysis form.

First, a circle with radius r is used as analysis circle. This circle is used over each point of the edge. After that, we enclose each contour point that is located within the circle with the center point of the circle. The connected pixels with the center pixel are ignored. This proposed process is illustrated in Fig. 16.

Figure 17 shows the application of the proposed join method on the edge image using a radius 10. The first image in Fig. 17 represents the edges. The second image illustrates the circle analysis of each edge point. Third image is the resulted image after applying the proposed join method.

We can notice that the thickness of some false edges is too important to achieve proper closure of edge. Thus, only the edges having a strength thickness are considered and used on the join edge method.

Figure 18 represents the obtained results of the join edge method proposed using different radius.

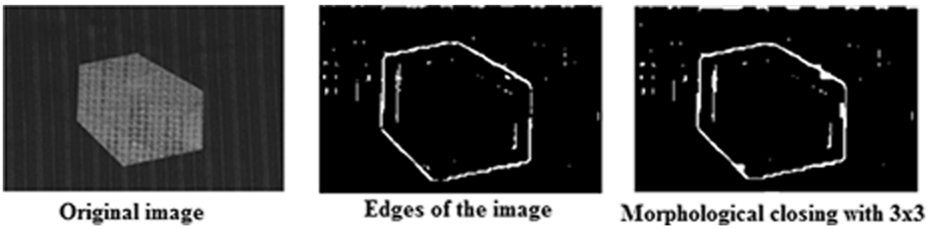


Fig. 15 Edges closing using Morphological closing

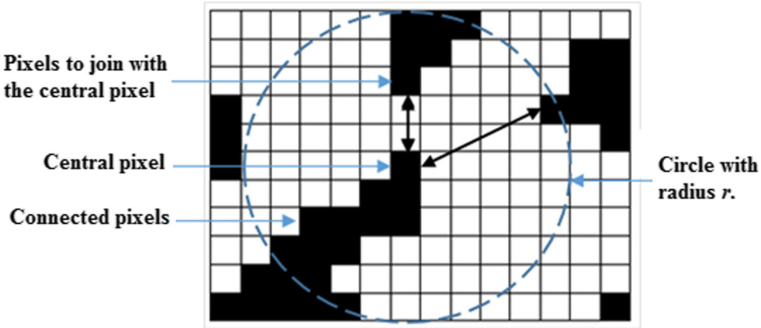


Fig. 16 Joining edge point method

Fig. 17 Application of the joining edge method (radius = 10)

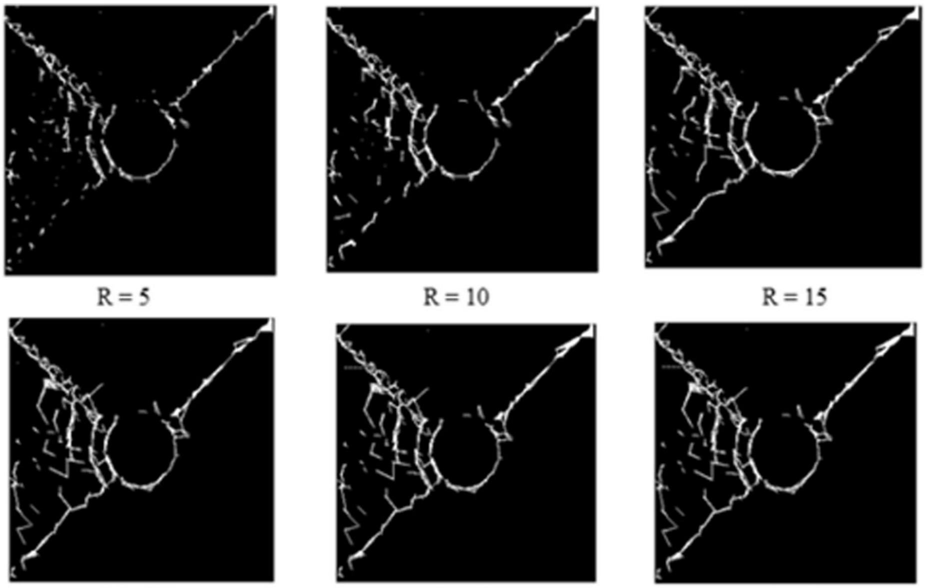


Fig. 18 Join edge method using different radius (5 to 30 pixels)

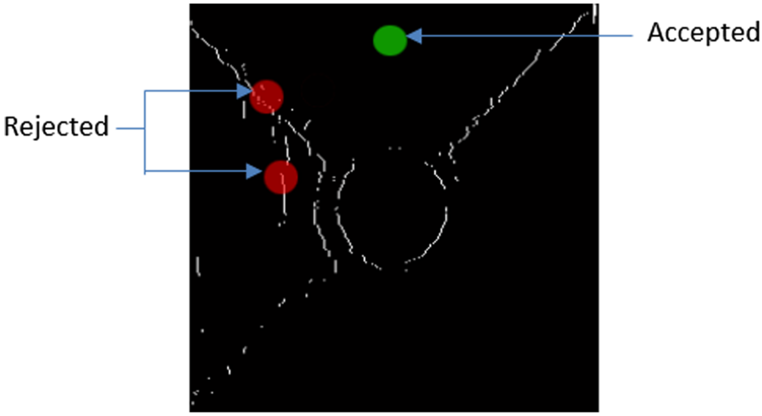


Fig. 19 The proposed growing region segmentation using circular shape

We can notice from Fig. 18 that the edges are closed but the obtained results are not good. Several false regions are generated using the join edge method. Thus, we have proposed a new method which is based on edges and regions to improve the segmentation. The proposed approach is explained in the next section.

The growing region segmentation In order to improve the join edge method, we propose to use a region segmentation method on the generated edge image. The main proposed idea is to cover the edge image obtained previously (after the pre-processing step) using a geometric shape (in our case we will use a circle with a radius R). Then, we keep only the forms (circles) which are not on an edge points. Any other shapes of circles which are located on an edge are rejected. The proposed method is illustrated in Fig. 19.

We applied the growing region segmentation on the edge image. We used circles with radius of 10 pixels Fig. 20.

The obtained image after selecting the relative circle can be segmented in different closed regions. All accepted and connected circles are regrouped in the same region. Thus, the ignored circles between different regions determine the boundaries of each region. Finally,

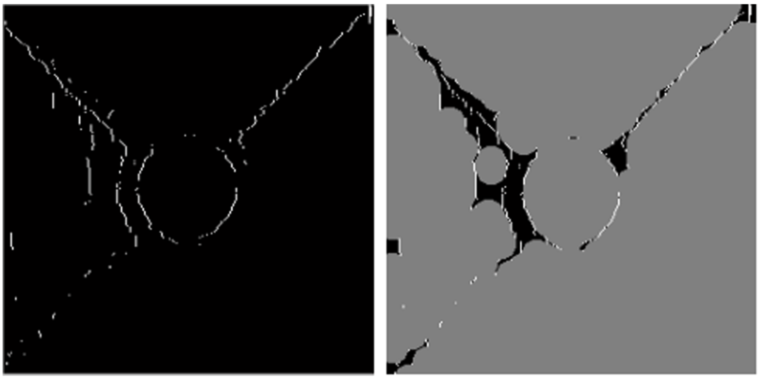


Fig. 20 Application of the growing region on the edge (joining edge method) segmented image

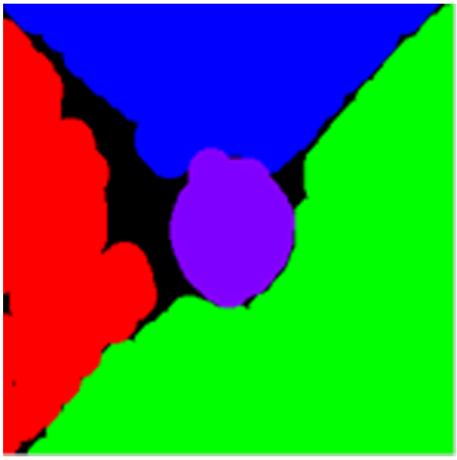


Fig. 21 Representation of different regions using different colors

in order to better differentiate the regions of textured images, we will label the different regions using different colors. The final obtained result is illustrated in Fig. 21.

The obtained results recognize well the different textures. But, there is clearly a border problem in this method (with black color). This problem is due to some small regions that do not belong to any region. Those small regions are illustrated in Fig. 21 by black pixels.

To solve this problem, we reduce the radius of the circles iteratively. Thus, regions with black color will be added iteratively to the recognized regions while reducing the radius of the circle. This iterative process and the obtained results are summarized in Fig. 22.

Figure 23 summarize each step of our proposed method.

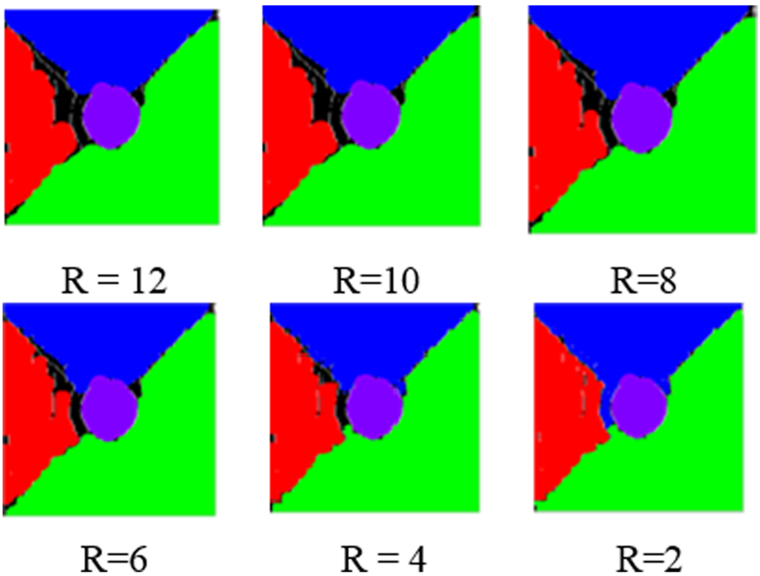


Fig. 22 Reducing the radius of the growing region method

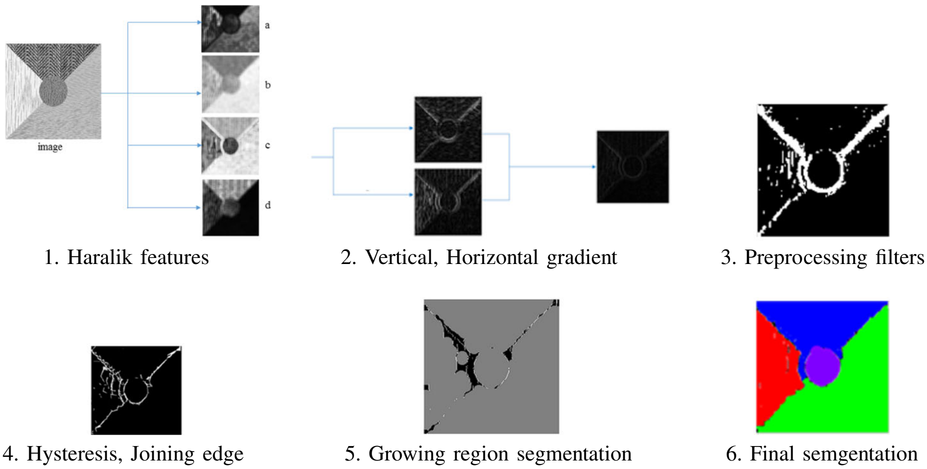


Fig. 23 Our proposed segmentation method

5 Experimental results

In this section, the evaluation of our proposed system is presented. In order to evaluate our proposed approach, a series of experiments using different textured images have been tested. The textured images used in the experiments are extracted from two famous texture image databases. The first database contains synthetic textured images from Brodatz data set [2] (that consist of two, three and five different textures). The second texture image database is presented by Oulu University (http://www.outex.oulu.fi/index.php, http://www.outex.oulu.fi/temp/orig.html). We have choose these two databases because they contain different and complex textures.

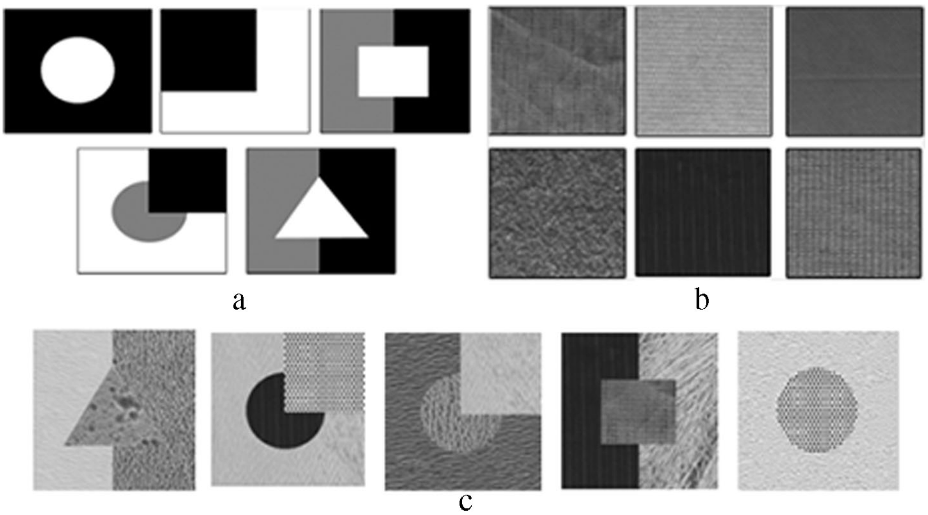
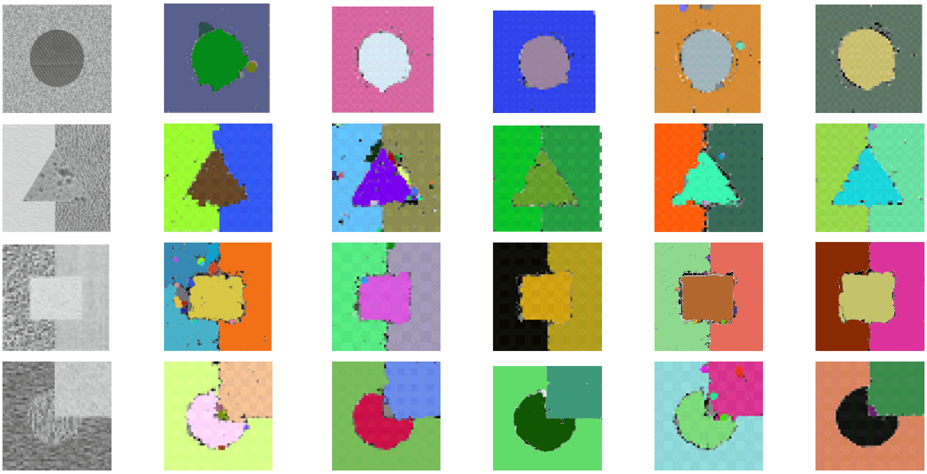


Fig. 24 **a** Models of textured images. **b** Textures. **c** Randomly generated images

Table 1 Results of proposed segmentation method using different preprocessing filters

Image	Without filter	Gaussian	Media	Nagao	Mean



Our test images are randomly generated from the two databases using different models. Some test images are illustrated in Fig. 24. The number and the type of the texture for each image is also randomly generated.

Figure 24c represents some generated test images. The test images are composed of naturel regular and random textures. The obtained segmentation results using the proposed system are illustrated in Table 1, where each region is represented by a uniform color. We illustrated also the impact of each filter on the segmentation.

As we can see in Table 1, using different filters improve the accuracy of the segmentation especially median and mean filters. Thus, those two filters are better to use with our proposed method.

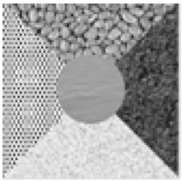
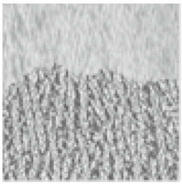
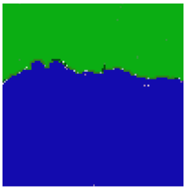
Table 2 illustrates the number of the recognized textures compared to the number of the different textures that are in the image. Textured images are generated randomly as shown in Fig. 24.

From our experiments we can notice that the Median filter gives the best results comapred to other filters. This because the median filter reduce the noise of the image [4] and makes the proposed method work efficiency.

Table 2 Number of recognized regions compared to the existing textures on test images

	None	Gaussian	Media	Nagao	Mean	Correct number
Texture 1	5	2	2	5	2	2
Texture 2	4	24	3	6	6	3
Texture 3	17	5	3	8	3	3
Texture 4	9	4	3	8	4	3
Texture 5	3	3	3	4	3	3
Texture 6	7	2	2	4	2	2

Table 3 Visual comparison between proposed method and method proposed in [18]

Textured image	Proposed method	Method [18]
		
		

In order to illustrate the improvement of our proposed approach compared to the previous research (which proposed the first database), we compare the performance of the proposed system with the method proposed by Yu Kian et al. [18]. Table 3 illustrated the obtained comparative results.

Quantitative matrices has been used to compare the segmentation architectures. First, we manually labelled test images and used as goal. Then, the obtained results of the segmentation method is converted into binary images. We used overlap ratio measure as similarity measure. The values range of the overlap ratio measure is between 0 (no overlap) and 1 (perfect overlap). First, the overlap ratio of each texture of one test image is calculated. After that, the average of all obtained overlap ratio measures is calculated and associated to this test image. Finally, the avrage of all images of each database [18] (http://www.outex.oulu.fi/index.php, http://www.outex.oulu.fi/temp/orig.html) is calculated.

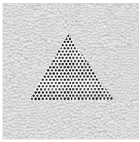
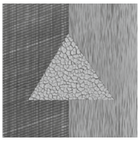
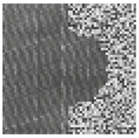
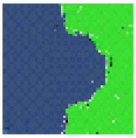
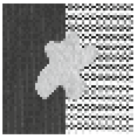
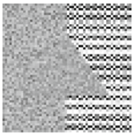
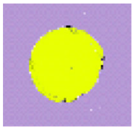
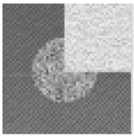
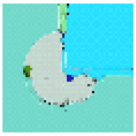
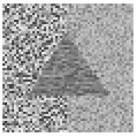
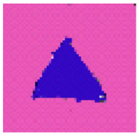
We compared three systems in our experiments. The system proposed by Jian Y. [18] (which use Gaussian Mixture Models and GLCM), second system proposed by Mohanty, A. K [25] (which proposed an image mining system based on GLCM and genetic association rule mining), and our proposed system. The three systems used the same feature descriptor (GLCM), that is why it is interesting to compare them. The obtained overlap ratio average of each database is illustrated in Table 4.

Segmentation methods give better result using database 1 because it contains syntetic texture (easy to recognize and segment) compared to the second database which contains natural textures. The system proposed by A. K [25] obtains bad results because the system is designed to recognize une object (flower). Thus, when we appyed it on images that contains

Table 4 Obtained overlap ratio average for each database using the three systems

Batbase	Jian Y. method [18]	A. K method [25]	Proposed method
Batabase 1 [18]	0.88	0.71	0.97
Batabase 2 (http://www.outex.oulu.fi/index.php, http://www.outex.oulu.fi/temp/orig.html)	0.79	0.65	0.92

Table 5 Some obtained experimental results using the proposed method

Image	Segmentation result	Image	Segmentation result
			
			
			
			

several textures (syntetic and natural) even using the training process of each texture (which take more times). The system proposed by Jian Y. [18] gives better results than A. K. This because the mean and variance value of gray level, Angular second moment and energy of the GLCM gives robustness to the system. The proposed method provide best results. The GLCM features, hysterisis threshold and the edge segmentation process (joining edge and growing region) localize and segment texture better. Other experimental results are illustrated in Table 5.

6 Conclusion

In this paper, we have proposed a new approach for texture segmentation. The proposed method is based on the GLCM texture features. After that, we proposed a new segmentation process.

Feature extraction: the GLCM is an efficient method to describe the texture but takes long calculation time. To improve this weakness of the GLCM we proposed to

- Reduce the size of the GLCM from 255 to 16.
- Fix the size of the analysis window to 15x15 pixels.
- Fix the offset of the analysis window to 1.

Preprocessing Filters: we have studied the impact of different filters on our proposed segmentation method. The median and mean filters reduce the noise and the false accepted edge. Then, we used the Hysteresis thresholding method instead a standard threshold, which is a dynamic threshold. After that, we used a morphological closing method to enclose the edges.

Joining the edge points: The small discontinuity of edges are joined in order to generate closed areas. We proposed a new method to join the closed edges, which recognize clearly main regions and detect strangely their boundaries. A circular geometric shape of the analysis window is used to extract efficiency the rounded edge. However, the generated edge image contains some false regions and edges.

The growing region segmentation: In order to improve the edge image and remove the false detected regions, we have proposed a growing region method to localize textures. We used a geometric shape (circle) to delineate areas of the image. This step is use to extract different textured regions of the image.

Experimental part illustrates the obtained results. This part shown the efficiency of the proposed approach compared to a previous work. Our proposed method have several parameters that may be changed depending on the nature of the texture (size and scale of the GLCM, applied filters, radius of the joining points method and the radius of the growing method). These parameters have been fixed based on the experiments.

In the future works, other geometric shapes will be applied to describe better the boundaries. We will add other texture analysis methods such as wavelet transform for the multi-resolution approach. This approche takes computational time because we used GLCM and different filters (depending on which filter is applied). We will also study the behavior and the robustness of the proposed approach on real images.

References

1. Biasotti S, Cerri A, Aono M, Hamza AB, Garro V, Giachetti A (2016) Retrieval and classification methods for textured 3D models: a comparative study. Vis Comput 32(2):217–241
2. Brodatz P (1966) Textures: A photographic album for artists and designers. Dover Publications, New York
3. Canny JF (1986) A computation approach to edge detectors. IEEE Trans Pattern Anal Mach Intell 8:34–43
4. Esakkirajan S, Veerakumar T, Subramanyam AN, PremChand CH (2011) Removal of high density salt and pepper noise through modified decision based unsymmetric trimmed median filter. IEEE Signal Process Lett 18(5):287–290
5. Hamouchene I (2014) Aouat S. A cognitive approach for texture analysis using neighbors-based binary patterns. IEEE 13th international conference on cognitive informatics and cognitive computing. IEEE, pp 94–99
6. Hamouchene I, Aouat S (2014) Texture matching using local and global descriptor. 5th European Workshop on Visual Information Processing (EUVIP). IEEE, 1–5
7. Hamouchene I, Aouat S (2014) A new texture analysis approach for iris recognition. AASRI Procedia 9:2–7
8. Hamouchene I, Aouat S (2014) Overview of texture analysis. International Journal of Computer Vision and Image Processing (IJCVIP) 4(2):32–51
9. Hamouchene I, Aouat S (2015) A new rotation-invariant approach for texture analysis. In: IFIP international conference on computer science and its applications. Springer, Cham, pp 45–53
10. Hamouchene I, Aouat S (2015) A new rotation-invariant approach for texture analysis. In: IFIP international conference on computer science and its applications. Springer, Cham, pp 45–53
11. Hamouchene I, Aouat S (2017) Rotation-invariant method for texture matching using model-based histograms and GLCM. Int J Reasoning-based Intell Sys 9(1):3–11
12. Hamouchene I, Aouat S (2017) A new approach for texture segmentation based on NBP method. Multimed Tools Appl 76(2):1921–1940
13. Hamouchene I, Aouat S (2019) Iris recognition system based on a new combined feature extraction method. Int J Adv Intell Paradig 14(1-2):80–106
14. Hamouchene I, Aouat S, Lacheheb HA (2013) A new segmentation architecture for texture matching using the LBP method. In: Science and information conference. IEEE, pp 411–415

15. Hamouchene I, Aouat S, Lacheheb H (2014) Texture segmentation and matching using LBP operator and GLCM matrix. Intelligent systems for science and information. Springer, Cham, pp 389–407
16. Haralick RM (1979) Statistical and structural approaches texture. Proc IEEE 67(5):786–804
17. Hojjatoleslami S, Kittler J (1998) Region growing: A new approach. IEEE Trans Image Process 7(7):1079–1084
18. Jian Y (2010) Texture image segmentation based on gaussian mixture models and Gray level co-occurrence matrix. International symposium on information science and engineering (ISISE), pp 149–152
19. Julesz (1975) Experiments in the visual perception of texture. Scientific American 232, bo. 4: 2–11
20. Kaganami HG, Beiji Z (2009) Region based segmentation versus edge detection. Intelligent information hiding and multimedia signal processing, pp 1217–1221
21. Kang WX, Yang QQ, Liang RR (2009) The comparative research on image seg-mentation algorithms. IEEE Conference on ETCS, pp 703–707
22. Krim H, Ben Hamza A (2015) Geometric methods in signal and image analysis. Cambridge University Press, Cambridge
23. Lerski R, Straughan K, Shad L, Boyce D, Bluml S, Zuna I (1993) Image texture analysis – An approach to tissue characterisation. Magnetic Res-onance Imaging, vol 11
24. Levine M (1985) Vision in man and machine. McGraw-Hill, New York
25. Mohanty AK, Bag A (2017) Image mining for flower classification by genetic association rule mining using GLCM features. Int J Adv Eng Manag Sci 3(5):239846
26. Nagao M, Matsuyama T (1979) Edge preserving smoothing. Journal Computer Graphics and Image Processing, vol 9
27. Otsu N (1979) A threshold selection method from Gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics, vol 9
28. Pratt W (1991) K. Wiley, Hoboken
29. Rajesh D, Swapna PD (2012) image segmentation techniques. IJECT 3(1)
30. Ramesh J, Rangachar K, Brian GS (1995) Machine vision, vol 5. McGraw-Hill, New York
31. Rastgarpour M, Shanbehzadeh J (2011) Application of AI techniques in medical image segmentation and novel categorization of available methods and tools. Proceedings of the international multi conference of engineers and computer scientists 1:190–202
32. Richards W, Polit A (1974) Texture matching. Kybernatic 16:155–162
33. Strzelecki M (1995) Segmentation of textured biomedical images using neural networks. PhD Thesis, Technical University of Łódź, Poland
34. Subramanya S, Teng JC, Fu Y (2002) Study of relative effectiveness of features in content-based image retrieval. 1st Inte Symp on Cyber Worlds, pp 168–175
35. Valkealathi K, Oja E (1998) Reduced multidimensional cooccurrence histograms in texture classification. IEEE Trans Pattern Anal Mach Intell 20(1):90–94
36. Yeh CH, Lin CY, Muchtar K, Lai HE, Sun MT (2017) Three-pronged compensation and hysteresis thresholding for moving object detection in real-time video surveillance. IEEE Trans Ind Electron 64(6):4945–4955
37. Yu H, He F, Pan Y (2018) A novel region-based active contour model via local patch similarity measure for image segmentation. Multimed Tools Appl 77(18):24097–24119
38. Yu H, He F, Pan Y (2019) A novel segmentation model for medical images with intensity inhomogeneity based on adaptive perturbation. Multimed Tools Appl 78(9):11779–11798
39. Zucker SW (1976) Toward a model of texture. Computer graphics image processing 5:190–202

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.