Grayscale Image Segmentation Based on Associative Memories

Enrique Guzmán Ramírez[1], Ofelia M. C. Jiménez[1], Alejandro D. Pérez[1], and Oleksiy Pogrebnyak[2]

[1] Universidad Tecnológica de la Mixteca, Oaxaca, Mexico
eguzman@mixteco.utm.mx
[2] Centro de Investigación en Computación, Instituto Politécnico Nacional, México D. F., Mexico
olek@pollux.cic.ipn.mx

Abstract. In this paper, a grayscale image segmentation algorithm based on Extended Associative Memories (EAM) is proposed. The algorithm is divided into three phases. First, the uniform distribution of the image pixel values is determined by means of the histogram technique. The result of this phase is a set of regions (classes) where each one is grouped into a certain number of pixel values. Second, the EAM training phase is applied to the information obtained at the first phase. The result of the second phase is an associative network that contains the centroids group of each of the regions in which the image will be segmented. Finally, the centroid to which each pixel belongs is obtained using the EAM classification phase, and the image segmentation process is completed. A quantitative analysis and comparative performance for frequently-used image segmentation by the clustering method, the *k*-means, and the proposed algorithm when it uses prom and med operators are presented.

Keywords. Image segmentation, associative memories, clustering techniques.

Segmentación de imágenes en escala de gris con base en memorias asociativas

Resumen. En este artículo, un algoritmo para segmentación de imágenes en tonos de gris con base en las Memorias Asociativas Extendidas (EAM) es propuesto. El algoritmo es dividido en tres fases, en la primer fase se determina una distribución uniforme de los valores de los pixeles de la imagen utilizando la técnica de histograma. El resultado de esta fase es un conjunto de regiones (clases) donde cada una agrupa un determinado número de valores de pixel. En la segunda fase se aplica el algoritmo de entrenamiento de las EAM sobre la información obtenida en la primera fase; el resultado de esta fase es una red asociativa que contiene los centroides de las regiones que serán usadas en la segmentación de la imagen. En la última fase, usando el algoritmo de clasificación de las EAM se obtiene el centroide al cual cada uno de los pixeles de la imagen pertenece y el proceso de segmentación es completado. En la sección de resultados se presentan un análisis cuantitativo y una comparativa de desempeño, utilizando imágenes estándares de prueba, entre nuestra propuesta y un algoritmo de segmentación, basado en técnicas de clasificación, frecuentemente utilizado, el algoritmo *k*-means.

Palabras clave. Segmentación de imágenes, memorias asociativas, técnicas de clasificación.

1 Introduction

Image segmentation is a fundamental process in many image, video, and computer vision applications. It is used to partition an image into separate regions which ideally correspond to different real-world objects. The importance of segmentation has long been recognized, but in the previous decades, a lack of good segmentation methods was one of many roadblocks towards making these applications feasible. Today, the problem of generating good segmentations is becoming increasingly critical because modern computer processors are able to provide the processing capabilities necessary to make many of these applications feasible. In applications such as content based image/video retrieval, computer vision for real-world computer interaction, and object- and content-based image/video compression, segmentation has become one of the most important problems that

must be solved in order to obtain successful results.

The goal of image segmentation is to find and isolate homogeneous regions contained in an image. For this purpose, some characteristics or computed properties such as color, intensity or texture are considered. The segmentation techniques can be based on a number of diverse methods: clustering methods, probabilistic methods, region growing methods, histogram-based methods, graph partitioning methods and neural networks-based methods. The central focus of this work is the image segmentation based on clustering methods. This kind of a method first divides the image into a small amount of regions and then assigns each image pixel to the region that presents the nearest matching based on similarity criterion. In clustering methods, the *k*-means algorithm is the most popular technique, and it is used to partition an image into *k* clusters [11, 12].

Cluster initialization is one of the major problems in k-means based algorithms: the final clustering may differ depending on the initial cluster center positions. To overcome this problem, *k*-means algorithm is run with multiple different initial partitions [8] that increase the computational complexity of the method. To overcome the shortcomings of the *k*-means technique, a number of clustering methods were proposed, we will mention some of them.

A popular technique for image segmentation derived from *k*-means method is the fuzzy c-mean (FCM) algorithm, which first was proposed in [5]. It performs fuzzy partitioning through an iterative optimization of the objective function, with the update of the cluster centers. Unfortunately, the algorithm does not incorporate any information about spatial context, which cause it to be sensitive to noise and imaging artifacts [1, 22]. To overcome the deficiencies of the base FCM algorithm, its various modifications were proposed that incorporate local spatial information as constraints in the objective function or impose a spatial penalty on the membership functions [22], Yan Ling Li and Yi Shen [21] presented an extended version of the robust fuzzy clustering method (RFCM) for image segmentation of noisy images [20]. This new algorithm is based on the kernel-induced distance measure which extends the RFCM algorithm to the corresponding kernelled version named "*Kernel Robust Fuzzy C Means*" (KRFCM) by the kernel methods. Furthermore, the KRFCM algorithm includes a class of robust non-Euclidean distance measures for the original data space to derive new objective functions and thus clustering the non-Euclidean structures in data.

Shinichi Shirakawa and Tomoharu Nagao proposed a method for evolutionary image segmentation based on multi-objective clustering [13]. First, this algorithm initializes the set of pixels using the minimum spanning tree (MST) technique. Next, by means of the multi-objective clustering with automatic k-determination (MOCK), developed by Handle and Knowles [6], [7], two complementary objectives based on cluster compactness and connectedness are optimized using a multi-objective evolutionary algorithm (MOEA) [3, 4]. In this case, the objectives are on overall deviation (measure of the similarity of pixels in the same region) and edge value (measure of the difference in the boundary between the regions). MOCK returns not just one solution, but an entire set of solutions. Finally, the Strength Pareto Evolutionary Algorithm 2 (SPEA2) is used to find the optimal solution.

In [19], the authors suggest a hybrid intelligent color image segmentation method. This work combines two segmentation approaches: region growing and clustering. Region growing is used to generate initial segmentation; it focuses on local variations of an image with fast speed. The final image segmentation is realized by the minimum spanning tree (MST) method which views every region produced by the region growing process as a node and it is used to extract the global property of the image, then a particle swarm optimization is applied to get the best threshold of MST and improve the algorithm speed.

In spite of the fact that *k*-means was first proposed over 50 years ago, it remains to be the most popular clustering algorithm for image segmentation due to its simplicity and ease of implementation [8] and many authors often use it as a reference method to compare the performance of different image segmentation techniques (for example, [11, 12]).

On the other hand, the image segmentation algorithm is an active part of systems that must be employed at real-time processing and to process a big quantity of information, for example, in an artificial vision system, then, it is necessary to propose algorithms of low computational complexity that allow a successful functioning of this type of applications.

In this paper, we propose an efficient grayscale image segmentation algorithm based on the Extended Associative Memories (EAM). The proposed algorithm is not iterative and therefore has low time computational complexity in comparison to the *k*-means method. It uses a simple histogram technique to initialize a set of classes (clusters). At the learning stage, the centroids of each of the regions in which the image will be segmented are determined, and the EAM classification phase completes the image segmentation process. The proposed algorithm is simple and has low computational complexity since its operation is based on morphological operations necessary for EAM learning and classification.

The remaining sections of this paper are organized as follows. In the next section, a brief theoretical background of EAM is given. In Section 3 we describe our proposal, namely, the grayscale image segmentation algorithm based on EAM. Numerical simulation results obtained for the conventional image segmentation techniques (*k*-means) and the proposed algorithm, when it uses **prom** and **med** operators, are provided and discussed in Section 4. Finally, Section 5 includes the conclusions of this paper.

2 Theoretical Background of Extended Associative Memories

An associative memory designed for pattern classification is an element whose fundamental purpose is to establish a relation of an input pattern $\mathbf{x} = [x_i]_n$ with the index i of a class c_i (see Fig. 1).

Let $\{(\mathbf{x}^1, c_1), (\mathbf{x}^2, c_2), ..., (\mathbf{x}^k, c_k)\}$ be k couples of a pattern and its corresponding class index defined as the *fundamental set of couples*, composed by a pattern and its corresponding class-index. The fundamental set of couples is represented by

$$\{(\mathbf{x}^\mu, c_\mu) \mid \mu = 1,2,...,k\} \tag{1}$$

where $\mathbf{x}^\mu \in \mathbf{R}^n$ and $c_\mu = 1,2,...,N$.

Fig. 1. Scheme of the associative memory for pattern classification

The associative memory is represented by a matrix generated from the fundamental set of couples y denoted by $\mathbf{M}$.

In 2004, Sossa et al. proposed an associative memory model for the classification of real-valued patterns [14]. This model, named Extended Associative Memories (EAM), is an extension of the Lernmatrix model proposed by K. Steinbuch [15]. The EAM is based on the general concept of the learning function of associative memory and presents a high performance in pattern classification of real value data in its components and with altered pattern version [1].

2.1 The EAM Training Phase

In order to understand the EAM training phase, it is necessary to define the element that allows relating the patterns with the class to which they belong.

Definition 1. Let $(c_1, c_2, ..., c_i)$ be a set of i classes, where each class groups a certain number of patterns, these elements form the fundamental set of couples. Then, there exist a function, denoted as ϕ, whose goal is to codify the information of all the patterns belonging to class i at the i-th row of the associative memory $\mathbf{M}$.

Now, the EAM training phase consists on evaluating the function ϕ for each class. Then, the matrix $\mathbf{M}$ can be structured as

$$\mathbf{M} = \begin{bmatrix} \phi_1 \\ \vdots \\ \phi_N \end{bmatrix} \quad (2)$$

where ϕ_i is the evaluation of ϕ for all patterns of class i, $i = 1,2,...,N$. The function ϕ can be evaluated in various manners. The arithmetical average operator (**prom**) is frequently used in the signals and images treatment. In [14], the authors studied the performance of this operator and the median operator (**med**) to evaluate the function ϕ**.** The goal of the training phase is to establish a relation between an input pattern $\mathbf{x} = [x_i]_n$, and the index i of a class c_i**.**

Let us consider that each class is composed for q patterns $\mathbf{x} = [x_i]_n$, and that $\phi_i = (\phi_{i,1},...,\phi_{i,n})$. Then, the training phase of the EAM, when the **prom** operator is used to evaluate the function ϕ_i, is defined as

$$\phi_{i,j} = \frac{1}{q}\sum_{l=1}^{q} x_{j,l}, \quad j = 1,...,n \quad (3)$$

The training phase of the EAM, when the **med** operator is used to evaluate the function ϕ_i, is defined as

$$\phi_{i,j} = \underset{l=1}{\overset{q}{\mathbf{med}}}\, x_{j,l}, \quad j = 1,...,n \quad (4)$$

The memory $\mathbf{M}$ is obtained after evaluating all functions ϕ_i. In the case when N classes exist and the vectors to classify are n-dimensional, the resultant memory $\mathbf{M} = [m_{ij}]_{N\times n}$ is:

$$\mathbf{M} = \begin{bmatrix} \phi_{1,1} & \phi_{1,2} & \cdots & \phi_{1,n} \\ \phi_{2,1} & \phi_{2,2} & \cdots & \phi_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{N,1} & \phi_{N,2} & \cdots & \phi_{N,n} \end{bmatrix} = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,n} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{N,1} & m_{N,2} & \cdots & m_{N,n} \end{bmatrix} \quad (5)$$

2.2 The EAM Classification Phase

The goal of the classification phase is the generation of the class index to which an input pattern belongs. The pattern classification by EAM is done when a pattern $\mathbf{x}^\mu \in \mathbf{R}^n$ is presented to the memory $\mathbf{M}$ generated at the training phase. The EAM classifies a pattern which is not necessarily one of those already used to build the memory $\mathbf{M}$.

When the **prom** operator is used, the class to which $\mathbf{x}$ belongs is given by

$$i = \underset{l}{\mathbf{arg}}\left[\overset{N}{\underset{l=1}{\wedge}}\ \overset{n}{\underset{j=1}{\vee}} \left| m_{lj} - x_j \right|\right] \quad (6)$$

In this case, the operators $\vee \equiv max$ and $\wedge \equiv min$ perform morphological operations on the difference of the absolute values of the elements m_{ij} of $\mathbf{M}$ and the component x_j of the pattern $\mathbf{x}$ to be classified.

When the EAM uses the **med** operator, the class to which $\mathbf{x}$ belongs is given by

$$i = \underset{l}{\mathbf{arg}}\left[\overset{N}{\underset{l=1}{\wedge}} \left| \underset{j=1}{\overset{n}{\mathbf{med}}}\, m_{lj} - \underset{j=1}{\overset{n}{\mathbf{med}}}\, x_j \right|\right] \quad (7)$$

In this case, the operator $\wedge \equiv min$ performs morphological erosion over the absolute difference of the median of the elements m_{ij} of $\mathbf{M}$ and the median of the component x_j of the pattern $\mathbf{x}$ to be classified.

Theorem 1 and Corollaries 1–5 from [14] govern the conditions that must be satisfied to obtain a perfect pattern classification. The pattern may be from the fundamental set of couples or it may be used in an altered version.

3 Grayscale Image Segmentation based on Extended Associative Memories

In this section, we introduce our proposal, the grayscale image segmentation algorithm based on EAM. The algorithm is divided into three phases.

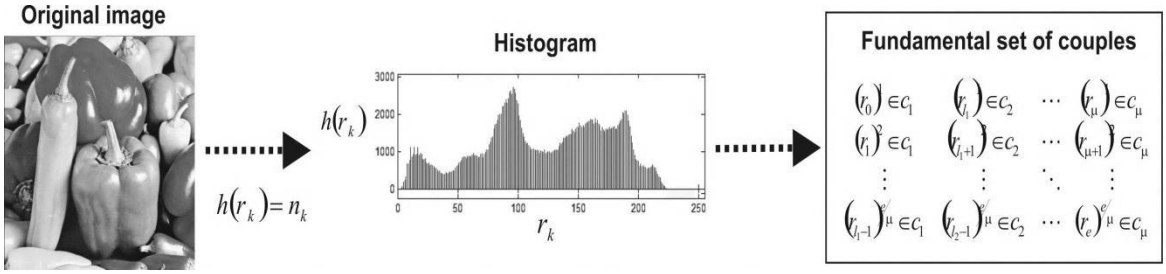

Fig. 2. Detecting the image features by means of histogram method

3.1 Proposed Algorithm

At the first phase, by means of the histogram technique, a set of regions (classes) is obtained, where each one groups a certain number of pixel values, these classes form the fundamental set of couples. That is to say, the goal of this phase is to group image pixels with similar value in a specific class. Now we explicate this process (see Fig. 2) First, the histogram of the image that will be segmented is obtained. The histogram of a digital image with gray levels in the range $[0, L-1]$ may be defined as a discrete function given by

$$h(r_k) = n_k \tag{8}$$

where $L = 2^m$, m is the number of bits/pixel, $k = 0,1,2,...,L-1$, r_k is the k-th gray level, and n_k is the number of pixels in the image having gray level r_k.

Now, the elements r_k where $h(r_k) = 0$ are excluded from the process. Then, the set of pixel values, denoted as X, is used to form the regions is defined as follows:

$$r_k \in X \rightarrow h(r_k) \neq 0 \tag{9}$$

The set X is composed of *e* elements, where $e \leq L$.

Next, the elements that will form each region must be defined. If in the segmentation process μ regions are defined, then the number of elements that will group each region is e/μ. Thus, the limits of regions are defined by $l_i = (i)(e/\mu)$ and the belonging of a pixel to a region is defined as follows:

$$\begin{aligned} &r_k \in c_1 \rightarrow r_k \leq l_1 \\ &r_k \in c_2 \rightarrow l_1 < r_k \leq l_2 \\ &r_k \in c_\mu \rightarrow l_{\mu-1} < r_k \leq e \end{aligned} \tag{10}$$

Finally, the fundamental set of couples, which tends to be a uniform distribution of the image pixel values, is defined as

$$\begin{matrix} (r_0)^1 \in c_1 & (r_{l_1})^1 \in c_2 & \cdots & (r_\mu)^1 \in c_\mu \\ (r_1)^2 \in c_1 & (r_{l_1+1})^2 \in c_2 & \cdots & (r_{\mu+1})^2 \in c_\mu \\ \vdots & \vdots & \ddots & \vdots \\ (r_{l_1-1})^{e/\mu} \in c_1 & (r_{l_2-1})^{e/\mu} \in c_2 & \cdots & (r_e)^{e/\mu} \in c_\mu \end{matrix} \tag{11}$$

We observe that each feature vector contains only one descriptor, that is $\mathbf{r}^j = [r_i]_p$, where $p = 1$; for simplicity we will denote it as r^j, $j = 1,2,...,e/\mu$

In the second phase, the region centers are generated applying the EAM training phase to the fundamental set of couples, defined by the equation (11).

Each element r that is a part of a class of the fundamental set of couples includes only one image characteristic, the intensity (gray level), thus the algorithm handles scalar numbers instead of vectors. Furthermore, each class is composed of e/μ elements r; therefore, the function that evaluates each class is $\phi_i = (\phi_1^i)$, $i = 1,2,...,\mu$. Due to this fact, the generalizing learning mechanism of the EAM, denoted by expressions (3) and (4), must be rewritten.

When the **prom** operator is used to generate $\mathbf{M}$, the expression (3) is rewritten as

$$\phi^i = m_i = \frac{\mu}{e}\sum_{j=1}^{e/\mu} r^j \qquad (12)$$

when **med** operator is used to generate $\mathbf{M}$, the expression (4) is rewritten as

$$\phi^i = m_i = \underset{j=1}{\overset{e/\mu}{\mathbf{med}}}\, r^j \qquad (13)$$

where $i = 1,2,..., \mu$.

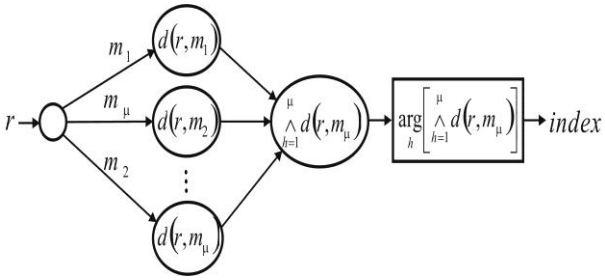

Fig 3. Structure of the associative memory for the grayscale image segmentation

Since μ classes or regions exist and the elements that integrate them contain only one descriptor, then the associative network that contains the centroids group of each of the regions in which the image will be segmented, is represented by the column vector $\mathbf{M} = [m_k]_\mu$:

$$\mathbf{M} = \begin{bmatrix} \phi^1 \\ \phi^2 \\ \vdots \\ \phi^\mu \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_\mu \end{bmatrix} \qquad (14)$$

The synaptic weight m_h represents the centroid of the region h, $h = 1,2,..., \mu$.

Finally, in the third phase where the classification phase of the EAM is applied, the region to which each image pixel belongs is obtained and the image segmentation process is completed.

In this case, each feature vector contains only one descriptor, denoted by r ($p = 1$), and μ regions have been defined by expression (11).

Then, the computation of $d(r, m_h)$ when the **prom** operator is used to generate the memory $\mathbf{M}$ is defined by $d(r, m_h) = \underset{j=1}{\overset{p}{\vee}} |m_{hj} - r_j|$; for this case $p = 1$, then

$$d(r, m_h) = |m_h - r| \qquad (15)$$

and, when the **med** operator is used, the memory $\mathbf{M}$ is defined by $d(r, m_h) = \left| \underset{j=1}{\overset{p}{\mathbf{med}}} m_{hj} - \underset{j=1}{\overset{p}{\mathbf{med}}} r_j \right|$, as $p = 1$, then

$$d(r, m_h) = |m_h - r| \qquad (16)$$

One can conclude that for both operators the degree of belonging of the pixel r to each one of the μ regions is defined by the same expression. The result is a μ-dimensional column vector

$$\begin{bmatrix} d(r, m_1) \\ d(r, m_2) \\ \vdots \\ d(r, m_\mu) \end{bmatrix}$$

Finally, the morphologic operator *min* is applied to this vector to establish to which class a new feature vector belongs. The class is indicated by the index of the row of $\mathbf{M}$ that presents the nearest matching with the feature vector r

$$index = centroid = \underset{h}{\mathbf{arg}} \left[\underset{h=1}{\overset{\mu}{\wedge}} d(r, m_\mu) \right] \qquad (17)$$

The segmentation process finishes when all pixels have been replaced by the centroid value of the class (index) to which it belongs. Since this algorithm is used to segment grayscale images and uses only one descriptor to form the feature vectors, it is the simplest case and, therefore, it offers the highest processing speed and the

lowest demand of system memory. Fig. 3 shows the structure of the associative memory generated to segment grayscale images when only one descriptor is used.

3.2 Complexity Analysis of Proposed Algorithm

In this subsection, we analyze both the time and space complexity of the proposed algorithm.

3.2.1 Time Complexity

In order to measure the algorithm time complexity, we have analyzed the number and type of arithmetical operations used by our algorithm during the image segmentation process of the image of $hi \times wi$ size in μ regions. Our proposal consists of three phases: generation of the fundamental set of couples, generation of the region centers and image pixel classification.

The first phase is based on the histogram technique; the number of operations that this phase needs to complete its function is defined as

$$\begin{aligned}&\left((hi \times wi) + \mu + L\right)\text{adds} + L\,\text{comparisons} \\ &+1\,\text{division}\end{aligned} \tag{18}$$

The second phase applies the EAM training phase to the fundamental set of couples; when the **prom** operator is used and based on the equation (12) implementation, the number of operations that this phase needs to complete its function is defined as

$$e\ \text{adds} + \mu\,\text{divisions} \tag{19}$$

where *e* is the number of elements that satisfy the condition $h(r_k) \neq 0$.

When the **med** operator is used and based on the equation (13) implementation, the number of operations that this phase needs to complete its function is defined as equation (20).

The third phase uses the classification phase of the EAM based on equations (15) and (16) implementations, the number of operations that this phase needs to complete its function is defined as eqation (21).

$$\begin{aligned}&\left(\mu \times \frac{e}{\mu}\right)\text{adds} + \mu\,\text{medians} \\ &= \left(\mu \times \frac{e}{\mu}\right)\text{adds} + \mu\left(1\,\text{add} + 1\,\text{shift}\right)\end{aligned} \tag{20}$$

$$\left(hi \times wi \times \mu\right)\left(1\,\text{sub} + 1\,\text{comparison} + 1\,\text{abs}\right) \tag{21}$$

Thus, the number and type of arithmetical operations that our algorithm with **prom** or **med** operators realizes to complete the image, with L gray levels, depend on the image size and the regions number, and the segmentation process is defined by equations (18), (19) and (21) when the **prom** operator is used and by equations (18), (20) and (21) when the **med** operator is used.

3.2.2 Space Complexity

The algorithm space complexity is determined by the amount of memory required for its execution. To segment an image of $hi \times wi$ size in μ regions, the proposed algorithm implementation, for **prom** operator, requires one matrix $\text{ImageSeg}\left[hi \times wi\right]$ and four vectors $\text{Hist1}[L]$, $\text{Hist2}[L]$, $\text{Regions}[\mu]$ and $\text{Mem}[\mu]$. Thus, the number of memory units (mu) required for this process is

$$\begin{aligned}&mu\_\text{ImageSeg} + mu\_\text{Hist1} + mu\_\text{Hist2} + \\ &mu\_\text{Regions} + mu\_\text{Mem} \\ &= (hi \times wi) + 2L + 2\mu\end{aligned} \tag{22}$$

For **med** operator, the proposed algorithm implementation needs the same elements that are used for **prom** operator and additionally a vector $\text{aux}[e/\mu]$; then, the mu required for this process is

$$\begin{aligned}&mu\_\text{ImageSeg} + mu\_\text{Hist1} + mu\_\text{Hist2} + \\ &mu\_\text{Regions} + mu\_\text{Mem} + mu\_\text{aux} \\ &= (hi \times wi) + 2L + 2\mu + (e/\mu)\end{aligned} \tag{23}$$

The variable ImageSeg stores the image to segment and the segmentation process result; the variables Hist1 and Hist2 store the histogram

result; the variable $\mathrm{Regions}$ stores the limits of the regions; the variable Mem stores the associative memory; the variable aux is used to compute the medians. For grayscale image (m bits/pixel), all variables store values that are within the range $[0, L-1]$, $L = 2^m$. Thus, for $m = 8$, all variables can be declared type *byte*.

4 Experimental Results and Discussion

In this Section, we present the experimental results obtained when the proposed algorithm is applied to grayscale images obtained from the Berkeley Segmentation Dataset [18]. For this purpose, a set of test images with 256 gray levels was used in simulations (see Fig. 4). The experiments include a comparative performance for frequently-used image segmentation by the clustering method, the *k*-means, and the proposed algorithm when it uses **prom** and **med** operators.

First, for a human visual evaluation (subjective analysis), the images segmentation results with several values of the parameter μ (clusters number) are presented. Second, an objective analysis of the performance evaluation of region segmentation algorithms based on a variant of ground-truth (GT) paradigm and making use of information theory concepts is presented. Finally, we present a complexity analysis for both proposed algorithm and k-means algorithm.

Our first experiment applies the two algorithms (*k*-means and our proposal with **prom** and **med** operators) on test images by modifying the parameter μ.

Figs. 5 shows the visual results of this process on ballerina, pigs, light-aircraft and vitnamita images.

The second experiment consists in an objective performance evaluation of region segmentation generated by the proposed algorithm. Typical objective functions in clustering formalize the goal of attaining high intra-cluster similarity (image objects within a cluster are similar) and low inter-cluster similarity (image objects from different clusters are dissimilar).

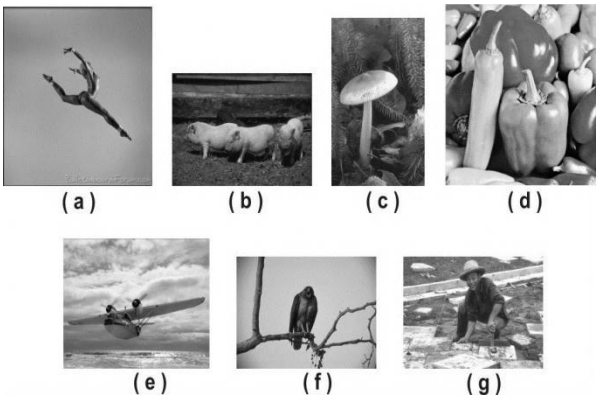

Fig. 4. Test images. (a) ballerina, (b) pigs, (c) mushroom, (d) peppers, (e) light-aircraft (f) hawk and (g) vitnamita

The proposed segmentation algorithm is based on a clasification method implemented by means of extended associative memories, therefore, we can consider the image segmentation problem as one of data clustering and, as a consequence, we can use the GT-paradigm for objective evaluation of the segmentation results.

For this purpose, we use a variant of GT-paradigm where an interest image was isolated (see Fig. 6).

When GT is available, segmentation can be evaluated in terms of *accuracy* and *robustness.* The accuracy reflects the precision of segmentation with respect to ground truth. The robustness is related to the accuracy degradation with respect to the degradation of the quality of test data.

A good segmentation evaluation should maximize the uniformity of pixels within each segmented region, and minimize the uniformity across the regions. Consequently, entropy-based criterions such as mutual information (MI) and normalized mutual information (NMI), measures of the disorder within a region, are natural characteristics to be incorporated into a segmentation evaluation method.

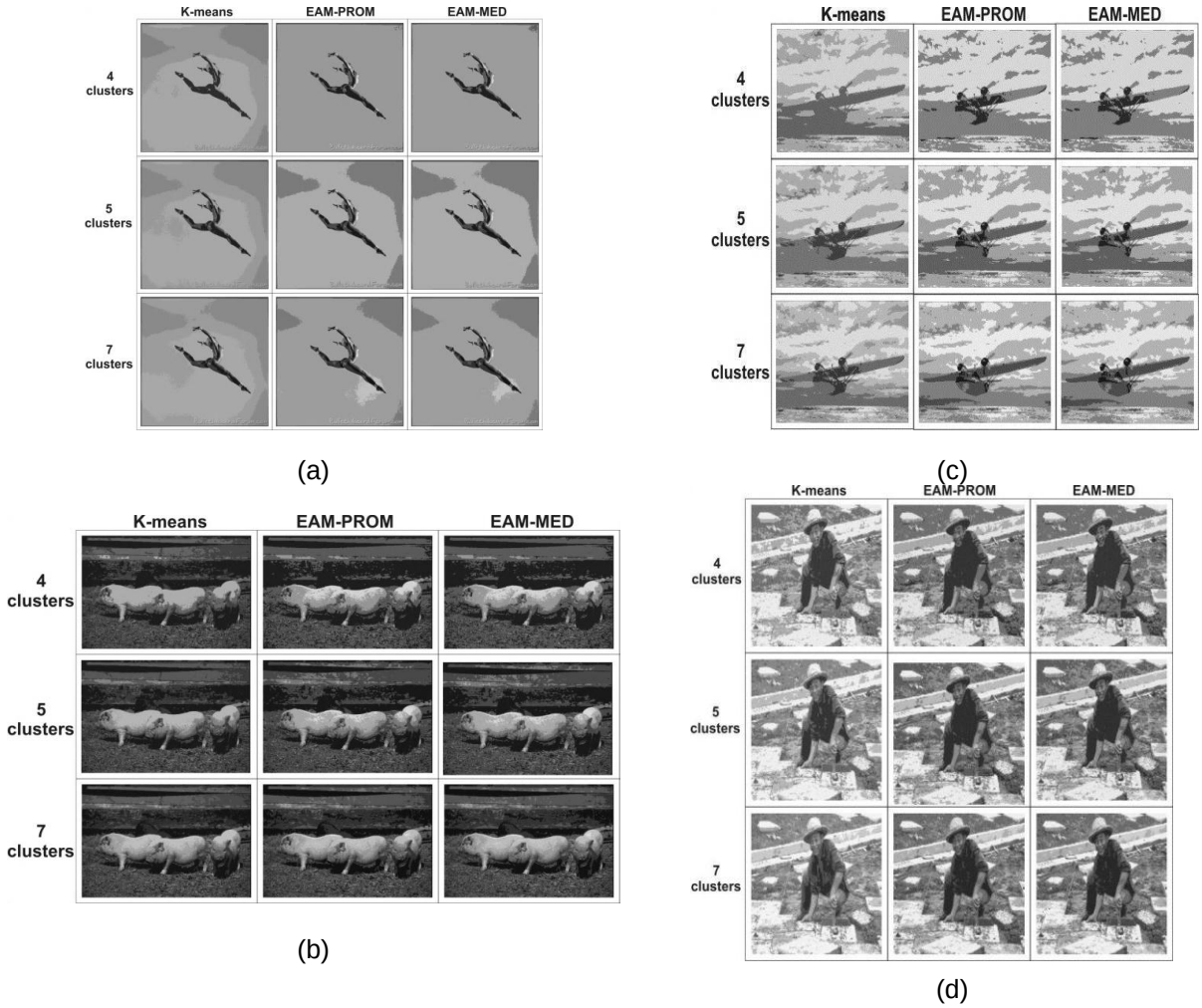

Fig. 5. Results of applying the k-means algorithm and the proposed algorithm, with **prom** and **med** operator, to pigs (a), ballerina (b), light-aircraft (c) and vitnamita (d) images

Recently, Xiaoyi Jiang *et al.* proposed to use MI and NMI to measure the existing similarity between the result generated by an image segmentation algorithm and it's GT [9, 10].

MI is a well-known concept in information theory. It is a symmetric measure that quantifies the mutual dependence between two random variables, or the information that they share. It measures how much the knowledge of one of these variables reduces our uncertainty about the other. This property suggests that the mutual information can be used to measure the information shared by two clusters, and thus, to assess their similarity.

If we consider X and Y as two images, MI can be defined as

$$\begin{aligned} I(X;Y) &= H(X) - H(X \mid Y) \\ &= H(Y) - H(Y \mid X) \\ &= H(X) + H(Y) - H(X,Y) \\ &= H(X,Y) - H(X \mid Y) - H(Y \mid X) \end{aligned} \tag{24}$$

where $H(X)$ and $H(Y)$ are the Shannon entropy of images X and Y respectively, $H(X|Y)$ denotes the conditional entropy, which is based on the conditional probabilities $p(x,y)$ and $H(X,Y)$ denotes the joint entropy.

Studholme *et al*. proposed a *normalized* measure of mutual information [16], which has a fixed lower bound of 0 and upper bound of 1. It takes the value of 1 when the two clusters are identical and 0 when the two clusters are independent, i.e. share no information about each other. The NMI is defined as

$$NMI(X;Y) = \frac{H(X)+H(Y)}{H(X,Y)} = \frac{I(X;Y)}{[H(X)+H(Y)]/2} = \frac{I(X;Y)}{\sqrt{H(X)H(Y)}} \quad (25)$$

In this second experiment we denoted the original image as $A=[a_{ij}]$. Also, we use the GT of the test images, these will be the X images (see Fig. 6).

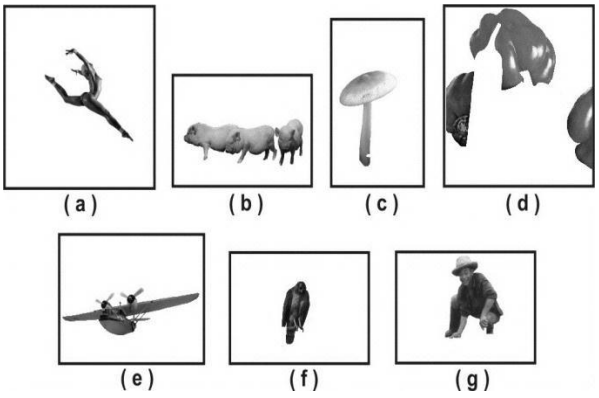

Fig. 6. GT Test images. (a) ballerina, (b) pigs, (c) mushroom, (d) peppers, (e) light-aircraft (f) hawk, (g) vitnamita

Next, we applied the two algorithms (k-means and our proposal with **prom** and **med** operators) on a test images with $\mu = 2$. Now, to isolate the corresponding information of the interest element and to obtain the Y image, we apply the following post-processing

$$y_{ij} = \begin{cases} a_{ij} & y_{ij} = PVIO \\ 255 & \text{other case} \end{cases} \quad (26)$$

where $PVIO$ is the acronym of Pixel Value of the Interest Object in the segmented image, $i = 1,2,...,\text{height image}$ and $j = 1,2,...,\text{width image}$.

Fig. 7 shows the resulting images when this process is applied in the images test.

To compare the performance of the algorithms, now we can compute the MI and NMI criteria between the X images (GT of the test images) and Y images (generated by k-means and our algorithm). Table 1 shows the obtained values for these criteria.

The equation $I(X;Y) = H(Y) - H(Y|X)$, where $H(Y)$ is the Shannon entropy of image Y, is computed on the probability distribution of the pixel values. $H(Y|X)$ denotes the conditional entropy, which is based on the conditional probabilities $p(y|x)$, the probability of pixel value y in gray scale image Y given the corresponding pixel in X has value x. When interpreting entropy as a measure of uncertainty, the equation $I(X;Y) = H(Y) - H(Y|X)$ is translated as "the amount of uncertainty about image Y minus the uncertainty about Y when X is known". In other words, MI is the amount by which the uncertainty about Y decreases when X is given: the amount of information X contains about Y. We use MI to quantify the information shared by the X and Y images. MI is a non-negative quantity where if $X = Y$ then $I(X;Y) = 0$.

The equation $I(X;Y) \le \mathbf{min}\{H(X), H(Y)\}$ indicate that the MI is upper bounded by both the entropies $H(X)$ and $H(Y)$. $H(X)$ and $H(Y)$ are based on the marginal probabilities $p(x)$ and $p(y)$ of the images X and Y respectively. Therefore, the MI upper limit is different for every image; in others words, the MI upper limit is a variable.

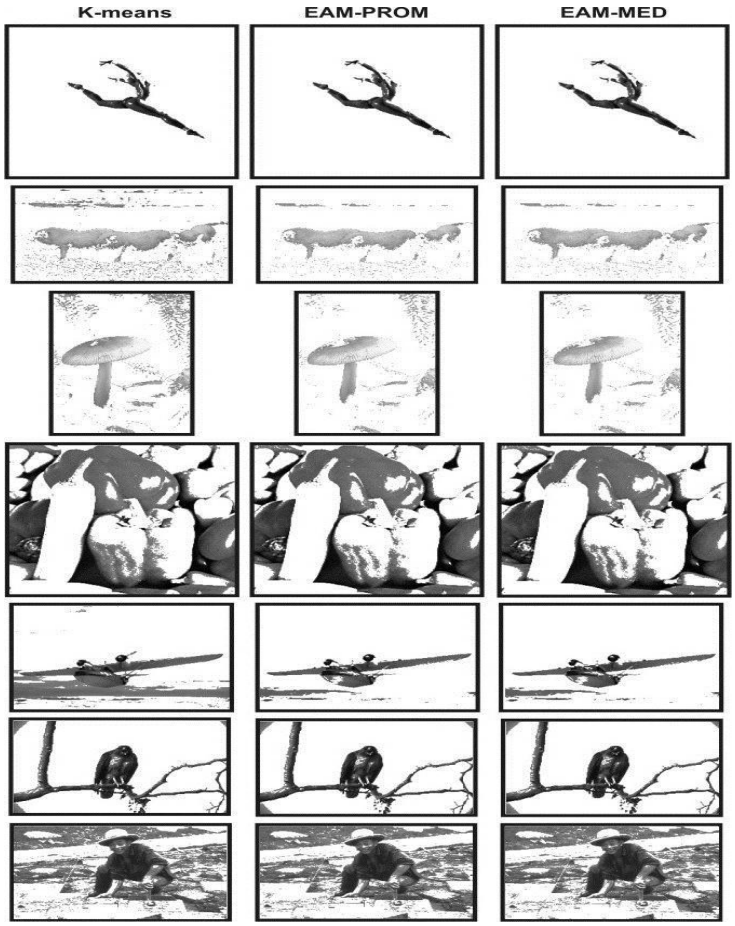

Fig. 7. Y Images obtained by applying the k-means and proposed algorithms on test images

Table 1. Obtained values of MI and NMI criterions

Image	Similarity measure	Algorithms		
		K-means	EAM-PROM	EAM-MED
ballerina	MI	0.521	0.565	0.565
	NMI	0.866	0.901	0.901
pigs	MI	1.373	1.212	1.212
	NMI	0.656	0.677	0.677
mushroom	MI	1.158	1.073	1.073
	NMI	0.641	0.684	0.684
peppers	MI	1.547	1.503	1.503
	NMI	0.565	0.572	0.572
light-aircraft	MI	0.984	0.859	0.859
	NMI	0.586	0.716	0.716
hawk	MI	0.683	0.679	0.679
	NMI	0.509	0.512	0.512
vitnamita	MI	1.316	1.335	1.335
	NMI	0.456	0.453	0.453

The NMI has a fixed bound 0 and 1. If $X = Y$ then $NMI(X;Y) = 1$.

The results obtained for both the MI and NMI (see Table 1) indicate that the Y images generated by our algorithm, when the **prom** or **med** operators are used, have a superior similarity with the GT test images that the images generated by the *k*-means algorithm.

In the last experiment, we compare the number and type of operations and the amount of memory used in our proposal and in the traditional image segmentation methods.

For this purpose, we implemented the image segmentation algorithm [17] that uses an iterative version of the k-means algorithm. Then, we analyzed its time and space complexity. The number and type of operations used by this algorithm to segment an image is defined as

$$\begin{aligned}(hi \times wi)(5\text{adds} + 3\text{comp} + \text{abs}) + \\ \mu(\text{mult} + \text{add} + \text{div}) + \\ itera[(hi \times wi)((\mu + 1)\text{add} + \\ \mu(\text{shift} + \text{comp} + \text{abs})) + \mu\,\text{div}]\end{aligned} \quad (27)$$

where $itera$ is the number of iterations that the k-means algorithm realizes to segment an image in μ regions.

Since this is an iterative algorithm, some operations will be repeated several times.

Now, to segment an image of $hi \times wi$ size in μ regions, the k-means algorithm implementation requires three vectors $\text{Hist1}[L]$, $\text{Hist2}[L]$, $\text{Regions}[\mu]$ and two matrix $\text{ImageSeg}[hi \times wi]$, $\text{Mask}[hi \times wi]$. Thus, the number of memory units (mu) required for this process is

$$\begin{aligned}mu\_\text{ImageSeg} + mu\_\text{Mask} + mu\_\text{Hist1} + \\ mu\_\text{Hist2} + mu\_\text{Regions} \\ = 2(hi \times wi) + 2L + \mu\end{aligned} \quad (28)$$

Based on the previous analysis, equations (18)-(23), (27) and (28), we can generate a comparative table of the operations and memory required by our proposal and the k-means algorithm in order to segment an image of size 512×512 pixels, with L=256 grayscale levels, in μ regions (see Table 2).

5 Conclusions

This paper introduces a novel clustering image segmentation algorithm based on extended associative memories. In the experimental result section, we presented both a subjective and quantitative analysis of our proposal and the traditional clustering segmentation method, the k-means. The quantitative performance evaluation is based on GT-paradigm and entropy criterions such as mutual information, normalized mutual information and variation of information. The results obtained from both the MI and NMI criterions indicate that the object extracted from the image by means of the segmentation process by our algorithm, when the **prom** or **med** operators are used, have a superior similarity with the GT test images in comparison to the one extracted by the *k*-means algorithm. Based on these results, we can conclude that our algorithm shows a better performance in the segmentation process than the traditional image segmentation clustering method, the *k*-means.

The proposed algorithm, when it uses the **prom** operator, has a low-computational complexity since its operation is based on applied morphological operations over the absolute values of differences; this fact results in a high processing speed.

Furthermore, when the extended associative memories are used as a classifier, they have the property (the ability) of replacing an input vector by the centroid index that presents the best similarity which is not altered by the fact that the vector has not been used in the centroids group generation. Since the group of centroids define the regions into which the image will be segmented, this ability of the EAM allows an efficient isolation of the objects of interest contained in the image.

Table 2. Computational complexity of both the proposed algorithm and the k-means algorithm when they are used to segment an image of size 512×512 pixels, with *l*=256 grayscale levels, in $\mu = 2$ regions

Algorithm	Iterations (*itera*)	Processing time (seconds)	Required memory (bytes)	Number and type of operations					
				comp	±	×	abs	shift	div
K-means	5	3.03	524802	3407872	2097154	2	2883584	2621440	4
Our proposal with **prom** operator		0.619	262660	524544	786818	--	524288	--	3
Our proposal with **med** operator		0.621	262788	524544	787204	--	524288	2	1

References

1. **Barron, R. (2006).** *Memorias asociativas y redes neuronales morfológicas para la recuperación de patrones.* Tesis de doctorado, Centro de Investigación en Computación, Instituto Politécnico Nacional, México, D.F.
2. **Beevi, S.Z., Sathik, M.M. & Senthamaraikannan K. (2010).** A Robust Fuzzy Clustering Technique with Spatial Neighborhood Information for Effective Medical Image Segmentation. *International Journal of Computer Science and Information Security*, 7(3), 132–138.
3. **Coello, C.A., Lamont, G.B. & Veldhuizen, D.A.V. (2007).** *Evolutionary Algorithms for Solving Multi-Objective Problems* (2nd ed.). New York: Springer.
4. **Deb, K. (2001).** *Multi-Objective Optimization Using Evolutionary Algorithms*. New York: John Wiley & Sons.
5. **Dunn J.C. (1973).** A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters, *Journal of Cybernetics,* 3(3), 32–57.
6. **Handl, J. & Knowles, J. (2004).** *Multiobjective clustering with automatic determination of the number of clusters* (Technical Report TR-COMPSYSBIO-2004-02). Manchester, UK, University of Manchester Institute of Science and Technology.
7. **Handl, J. & Knowles, J. (2007).** An evolutionary approach to multiobjective clustering. *IEEE Transactions on Evolutionary Computation*, 11(1), 56–76.
8. **Jain A.K. (2010)**. Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*, 31(8), 651–666.
9. **Jiang, X., Marti, C., Irniger, C. & Bunke, H. (2005).** Image Segmentation Evaluation by Techniques of Comparing Clusterings. *Image Analysis and Processing ICIAP 2005. Lecture Notes in Computer Science*, 3617, 344–351.
10. **Jiang, X., Marti, C., Irniger, C. & Bunke, H. (2006).** Distance Measure for Image Segmentation Evaluation, *EURASIP Journal on Applied Signal Processing*. 2006, 1–10.
11. **Ristic, D.M., Pavlovic, M. & Reljin, I. (2008).** Image Segmentation Method Based on Self Organizing Maps and K-Means Algorithm. *9th Symposium on Neural Network Applications in Electrical Engineering*, Belgrade, Serbia, 27–30.
12. **Samma, A.S.B. & Salam, R.A. (2009).** Adaptation of K-Means Algorithm for Image Segmentation. *World Academy of Science, Engineering and Technology* 50, 58–62.
13. **Shirakawa, S. & Nagao, T. (2009).** Evolutionary Image Segmentation Based on Multiobjective Clustering. *IEEE Congress on Evolutionary Computation*, Trondheim, Norway, 2466–2473.
14. **Sossa, H., Barrón, R. & Vázquez, R.A. (2004).** Real-valued Patterns Classification based on Extended Associative Memory. *Fifth Mexican International Conference on Computer Science.* Colima, México. 213–219.

15. **Steinbuch, K. (1961).** Die Lernmatrix. Biological Cybernetics, 1(1), 36–45.
16. **Studholme C., Hill, D.L.G. & Hawkes, D.J. (1999).** An overlap invariant entropy measure of 3D medical image alignment. *Pattern Recognition*, 32(1), 71–86.
17. **Tatiraju, S. & Mehta, A. (2008).** *Image Segmentation using k-means clustering, EM and Normalized Cuts* (Technical Report). University Of California, Irvine. http://www.ics.uci.edu/~dramanan/teaching/ics273a_winter08/projects/avim_report.pdf
18. The Berkeley Segmentation Dataset and Benchmark. (s.f.). Retrieved from http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/
19. **Xue-xi, Z. & Yi-min, Y. (2008).** Hybrid Intelligent Algorithms for Color Image Segmentation. *Chinese Control and Decision Conference*. Yantai, China. 264–268.
20. **Yang, Z., Chung, F.L. & Shitong, W. (2009).** Robust fuzzy clustering based image segmentation, *Applied Soft Computing Journal,* 9(1), 80–84.
21. **Yanling, L. & Yi, S. (2008).** Robust Image Segmentation Algorithm Using Fuzzy Clustering Based on Kernel-Induced Distance Measure. *International Conference on Computer Science and Software Engineering 2008*, Wuhan, China, 1, 1065-1068.
22. **Yanling, L. & Yi, S. (2010).** Fuzzy c-means clustering based on spatial neighborhood information for image segmentation. *Journal of Systems Engineering and Electronics,* 21(2), 323–328.

Enrique Guzmán Ramírez obtained an Electronic Engineers degree from Escuela Superior de Ingeniería Mecánica y Eléctrica, IPN, in 1992. Later he obtained MSc degree in Computer Engineering, Digital Systems specialty, in 2003, and PhD degree in Computer Science, in 2008, all degrees are from the Centro de Investigación en Computación, IPN. At present, he is a Research Professor at the Universidad Tecnológica de la Mixteca. His research interests include development and implement hardware architectures on reconfigurable logic for control, artificial neural networks and image processing applications.

Ofelia María del Carmen Jiménez Peña received her Bachelor's degree in Computer Engineering from Universidad Tecnológica de la Mixteca. Her research interests include development of image processing algorithms based on associative memories and artificial neural networks.

Alejandro D. Pérez received a Bachelor's degree in Computer Engineering from Universidad Tecnológica de la Mixteca. His research interests include image processing, associative memories and artificial neural networks.

Oleksiy Pogrebnyak received a Ph.D. degree from Kharkov Aviation Institute (now National Aerospace University), Ukraine, in 1991. Currently, he works at the Center for Computing Research of National Polytechnic Institute, Mexico. His research interests include digital signal/image filtering and compression, remote sensing.

Article received on 11/12/2010; accepted 05/02/2011.