Fuzzy Cognitive Maps Applied to Computer Vision Tasks

Gonzalo Pajares, María Guijarro, P. Javier Herrera, José J. Ruz,
and Jesús M. de la Cruz

Abstract. Computer vision is an emerging area which is demanding solutions for solving different problems. The data to be processed are bi-dimensional (2D) images captured from the tri-dimensional (3D) scene. The objects in 3D are generally composed of related parts that joined form the whole object. Fortunately, the relations in 3D are preserved in 2D. Hence, we can exploit this fact by considering specific and basic elements which are related to other elements in the 2D images. The relations with other elements allow establishing a link among them. Hence, we have the necessary ingredients to build a structure under the Fuzzy Cognitive Maps (FCMs) paradigm. FCMs have been satisfactorily used in several areas of computer vision including: pattern recognition, image change detection or stereo vision matching. In this chapter we establish the general framework of fuzzy cognitive maps in the context of 2D images and describe three applications in the three mentioned areas of computer vision. We also give some details about the performance of this paradigm in these applications.

1 Introduction

Computer vision is an emerging area which is demanding solutions for solving different problems. The basic data are bi-dimensional (2D) images captured from the three-dimensional (3D) real world.

Gonzalo Pajares
Dpto. Ingeniería del Software e Inteligencia Artificial, Facultad de Informática, Universidad Complutense, 28040 Madrid, Spain
e-mail: pajares@fdi.ucm.es

María Guijarro
Centro Superior de Estudios Felipe II, Ingeniería Técnica en informática de Sistemas 28300 Aranjuez, Madrid, Spain
e-mail: mguijarro@cesfelipesegundo.com

P. Javier Herrera · José J. Ruz · Jesús M. de la Cruz
Dpto. Arquitectura Computadores y Automática, Facultad de Informática, Universidad Complutense, 28040 Madrid, Spain
e-mail: pjherrera@pdi.ucm.es, {jjruz,jmcruz}@dacya.ucm.es

M. Glykas (Ed.): Fuzzy Cognitive Maps, STUDFUZZ 247, pp. 259–289, 2010.
springerlink.com

A look to the 3D world allows observing objects in the scene which are formed by structures or elements grouped together. Fortunately when the objects are mapped from the 3D world to the 2D image they preserve this grouping.

From this observation, an important issue addressed in computer vision tasks, based on image applications, is referred to how sensory devices perceive the objects in the scene, i.e. how the scene analysis problem is addressed. To deal with real-world scenes some criterion for grouping elements or features in the scene is required. In the work of Wang (2005) a list of major grouping principles is exhaustively studied. They are inspired in the Gestalt's principles (Koffka, 1935). The most important, from the point of view of computer vision tasks, are: *proximity*, labeled features that lie close in space tend to group; *similarity*, labeled features with similar properties or attributes tend to group; *connectedness*, labeled features that lie inside the same connected region tend to group.

These principles allow defining a spatial neighborhood in the 2D images which is a projection from the 3D scene. Now the problem is to build some structure where the above principles are to be mapped. Several approaches can be used, the Fuzzy Cognitive Maps (FCMs) is one of them.

Indeed, as explained in the next section, a FCM is a network of concepts where each concept represents some feature in the image. The concepts are joined by links representing some kind of relations among the features.

The application of FCMs for computer vision tasks requires the identification of the features to be represented as concepts; each concept should be characterized by a property, quantified by a specific value. Moreover, the concepts are linked among them and the strength of each link is also characterized by a value, which is determined by the application of the Gestal's principles.

This chapter describes the application of FCMs to three well addressed computer vision tasks, namely: 1) classification, 2) image change detection and 3) stereovision matching. The features used in this chapter are pixels for the tasks 1) and 2) and edge segments for the task 3). Other features (corners, regions) should be possible depending on the application.

Through this chapter we shall be concerned with the definition of the FCM framework for computer vision. Afterwards, we apply this framework for the three applications mentioned above. Much of our attention will be devoted to studying the method for finding concepts and relations between them. Some details about the performance of FCM on these applications are also provided.

2 Fuzzy Cognitive Maps: Topology and Basic Concepts

Fuzzy Cognitive Maps is a well developed modelling methodology for complex systems originating that allow to describe the behaviour of a system in terms of concepts. They have been used as a decision modelling tool under different approaches including those related with enterprises orientations (Xirogiannis et al., 2008; Xirogiannis and Glykas, 2004), e-business (Xirogiannis and Glykas, 2007) or financial applications among others (Xirogiannis and Glykas, 2005). Under its

most general approach, each concept represents an entity, a state, a variable or a feature of the system (Tsardias and Margaritis, 1997, 1999; Kosko, 1986, 1992; Miao and Liu, 2000). According to Kosko (1992), FCMs are fuzzy signed directed graphs with feedback. The directed edge e_{ik} from causal concept C_i to concept C_k measures how much C_i causes C_k. The edges e_{ik} take values in the fuzzy causal interval $[-1,+1]$, $e_{ik} = 0$ indicates no causality; $e_{ik} > 0$ indicates causal increase, this means that C_k increases as C_i increases and vice versa, C_k decreases as C_i decreases; $e_{ik} < 0$ indicates causal decrease or negative causality, C_k decreases as C_i increases and C_k increases as C_i decreases.

Given an FCM with a number n of concepts C_i, i.e. $i = 1,\ldots,n$, the value assigned to each concept, called activation level, can be updated iteratively, until convergence, based on the external influences exerted by the other nodes C_k on C_i and its self-influence through C_i. Several approaches have been proposed to map these influences, including those averaging both the internal and the external ones (Xirogiannis et al., 2008; Xirogiannis and Glykas, 2007, 2005, 2004). In this work we have chosen the proposed by Tsardias and Margaritis (1997, 1999) according to equation (1), because it includes a decay factor, explained below, which introduces a mechanism for achieving high stability in the network.

$$A_i(t+1) = f\left(A_i(t), \sum_{k=1}^{n} e_{ki}(t)A_k(t)\right) - d_i A_i(t) \tag{1}$$

The explanation of the terms in the equation (1) is as follows:

1. $A_i(t)$ and $A_k(t)$ are respectively the activation levels of the concepts C_i and C_k at the iteration, t. The sum is extended to all n concepts available. Nevertheless, only the concepts with edge values different from zero exert influences over the concept C_i trying to modify its current activation level $A_i(t)$ towards $A_i(t+1)$.
2. e_{ki} (t) are the fuzzy causalities between concepts, defined as above but considering that they could vary dynamically with the iterations.
3. $d_i \in [0,1]$ is the decay factor of certainty concept C_i. It determines the fraction of the current activation level that will be subtracted from the new one as a result of the concept's natural intention to get stable activation levels. The bigger the decay factor, the stronger the decay mechanism. This factor was introduced in Tsardias and Margaritis (1997, 1999) as a mechanism for introducing a degree of instability, so that those concepts destabilised intentionally but with a high degree of real stability tend towards its stabilized activation level. On the contrary, if the activation level is unstable, the decay mechanism induces a continuous variability on the activation level.
4. f is a non-linear function that determines the activation level of each concept. Some common choices of f are the following:
   a) sigmoid (Haykin, 1994),

$$f(x) = \tanh(x) \tag{2}$$

b) logistic (Haykin, 1994),

$$f(x) = \left(1 - e^{-x}\right)^{-1} \tag{3}$$

c) the function defined in the MYCIN expert system for the aggregation of the certainty factors (Shorliffe, 1976; Tsardias and Margaritis 1998)

$$f(x,y) = \begin{cases} x + y(1-x) & \text{if } x, y \geq 0, \\ x + y(1+x) & \text{if } x, y < 0, \\ (x+y)\Big/\left(1 - \min\left(|x|,|y|\right)\right) & \text{else} \end{cases} \tag{4}$$

where $|x|,|y| \leq 1$

The variable x in (2) and (3) represents a combination of $A_i(t)$ and $\sum_{k=1}^{n} e_{ki}(t)A_k(t)$, some times as a weighted average as mentioned before. In (4) $x = A_i(t)$ and $y = \sum_{k=1}^{n} e_{ki}(t)A_k(t)$.

The inference mechanism of FCMs can be summarized as follows: first, the FCMs are initialized. The activation level of each concept is set to a value based on a specific mechanism that must be defined. Because the various concepts are free to interact (Yaman and Polat, 2009), the activation of one concept influences the concepts to which it is connected. This interaction continues until:

1) a fixed-point equilibrium is reached.
2) a limit cycle is reached or equivalently a number of iterations.
3) chaotic behavior is exhibited (Tsardias and Margaritis, 1997).

3 Application of FCMs to Image Classification

This application is based on the work of Pajares et al. (2009). It describes a combined classifier where each pixel in the image must be classified as belonging to a cluster. There are different approaches for classification, two of the most popular are Fuzzy Clustering (*FC*) (Zimmermann, 1991) and the parametric Bayesian (*BP*) one (Duda et al., 2000). These classifiers perform a training phase and then make a decision based on this training. Actually, the tendency for making better decisions is the combination of classifiers (Kuncheva, 2004). Here, in the combination of *FC* and *BP*, is where the FCMs paradigm is applied.

We briefly describe both the training and decision phases which are concerning the simple classifiers. During the decision one, we will identify the pixels as features, which define the causal concepts, their activation levels, the causalities between concepts and the decay factor. The final goal is to classify the pixels as belonging to a cluster. This is carried out by combining the supports provided by the simple classifiers. These supports are membership degrees and probabilities provided by *FC* and *BP* respectively. The combination is made by defining a coefficient which can cope with both supports. This coefficient is part of the causal edges between concepts.

3.1 Training Phase

We start with the observation of a set X of N training samples, i.e., $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\} \in \Re^p$, where p is the data dimensionality. Each sample belongs to a known cluster w_j, where the number of clusters is c, i.e., $j = 1, 2, \ldots, c$. Each training sample is described by its vector $\boldsymbol{x}_i$ representing a pixel in the image, where its components are the three spectral RGB values of that pixel at the image location (x,y); hence in this example $p = 3$.

Under the above assumption, we know the distribution of the samples into the c clusters: this fact makes the training process supervised. For combination purposes, we apply two training procedures based on both *FC* and *BP* as described below.

3.1.1 Training through the Fuzzy Clustering

This process receives the input training patterns $\boldsymbol{x}_i$ and computes for each one, at the iteration t, its degree of membership μ_i^j in the cluster w_j and also the cluster centres v_j according to equation (5).

$$\mu_i^j(t+1) = \frac{1}{\sum_{r=1}^{c} \left(d_{ij}(t) / d_{ir}(t) \right)^{2/(b-1)}} \tag{5}$$

where $d_{ij}^2 \equiv d^2\left(\boldsymbol{x}_i, \boldsymbol{v}_j\right)$ is the squared Euclidean distance. The number $b > 1$ is called the exponential weight (Bezdek, 1981; Duda et al., 2000). The stopping criterion of the iteration process is achieved when $\left\| \mu_i^j(t+1) - \mu_i^j(t) \right\| < \varepsilon \;\; \forall ij$ or a number t_{max} of iterations is reached. Note, that the number of iterations for estimating the membership degrees and the cluster centers is different from the iterations in the *FCM*s, that appear in equation (1).

3.1.2 Training through the Parametric Bayesian Classifier

Following Duda et al. (2000), the Bayesian's classifier makes its decision about any pattern $\boldsymbol{x}$, which is to be classified, according to the following rule,

$$\boldsymbol{x} \in w_j \;\text{ if }\; p\left(\boldsymbol{x} \mid w_j\right) P(w_j) > p\left(\boldsymbol{x} \mid w_h\right) P(w_h) \quad \forall h \neq j \tag{6}$$

$P(w_j)$ and $P(w_h)$ are the prior probabilities that $\boldsymbol{x} \in w_j$ and $\boldsymbol{x} \in w_h$ respectively; $p\left(\boldsymbol{x} \mid w_j\right)$ and $p\left(\boldsymbol{x} \mid w_h\right)$ are the likelihoods that $\boldsymbol{x} \in w_j$ and $\boldsymbol{x} \in w_h$ respectively. Because we do not have prior information about the samples $\boldsymbol{x}$, which are to be classified, we assume that $P(w_j) = P(w_h)$. This means that the decision is made based only on the likelihoods, generally modelled as normal (Gaussian) probability density functions, which are obtained for each cluster w_j as follows,

$$p\left(\boldsymbol{x} \mid w_j\right)=\frac{1}{(2\pi)^{p/2}\left|\Sigma_j\right|^{1/2}}\exp\left[-\frac{1}{2}\left(\boldsymbol{x}-\boldsymbol{m}_j\right)^T\Sigma_j^{-1}\left(\boldsymbol{x}-\boldsymbol{m}_j\right)\right] \tag{7}$$

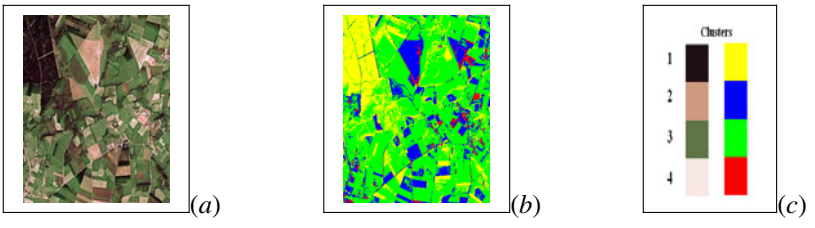
p is as before the data dimensionality, i.e., $p = 3$.

The goal of the training process for this classifier is to estimate both parameters: the mean $\boldsymbol{m}_j$ and the covariance Σ_j parameters, both for each cluster w_j with n_j samples belonging to such cluster. This is carried out through maximum likelihood estimation considering the distribution of the training samples into clusters, as follows,

$$\boldsymbol{m}_j=\frac{1}{n_j}\sum_{k=1}^{n_j}\boldsymbol{x}_k \qquad \Sigma_j=\frac{1}{n_j-1}\sum_{k=1}^{n_j}\left(\boldsymbol{x}_k-\boldsymbol{m}_j\right)\left(\boldsymbol{x}_k-\boldsymbol{m}_j\right)^T \tag{8}$$

where T denotes transpose.

Figure 1(*a*) displays an original training image of a 3D aerial scene captured with an airborne sensor, (*b*) the distribution of the samples in the four clusters according to the representation given in (*c*). In the left column of (*c*) the spectral signature is represented for each cluster centre, computed according to equation (5), i.e. through the *FC* classifier; in the right column, the label assigned to each cluster.



(*a*) (*b*) (*c*)

Fig. 1 (*a*) Original training image; (*b*) samples classified in four clusters as displayed in (*c*)

3.2 Decision Phase: Fuzzy Cognitive Maps

After the training, each new incoming sample $\boldsymbol{x}_s \in \Re^p$ must be classified as belonging to a cluster w_j based on the learning and knowledge achieved by the system during the training phase. This sample, like each training sample, represents a pixel at the image location (x,y) with its corresponding R,G,B spectral components. *FC* computes the membership degrees for $\boldsymbol{x}_s$ to each cluster trough the equation (5) and classifies the pixel according to the following rule: $\boldsymbol{x}_s \in w_j$ if $\mu_s^j > \mu_s^h$ for all $h \neq j$. *BP* computes the probabilities through equation (7); these probabilities can be used for classifying $\boldsymbol{x}_s$ according to the rule in the equation (6). Given an image to be classified, we compute these membership degrees and probabilities for each pixel in the image before the *FCM* structure is built. The

membership degrees are used as the initial activation levels for the *FCM* as described in the next section. In what follows, we define the network topology of the *FCM*, the equivalences between this specific *FCM*s derived form the general *FCM* framework, synthesized in equation (1). Afterwards, we combine probabilities and membership degrees for defining the directed dynamical $e_{ik}(t)$ causal edges between concepts and the decay factors, as required by the *FCM*.

3.2.1 Network Topology of the Fuzzy Cognitive Maps

For each cluster w_j, we build a network of concepts, net_j, where the topology of this network is established by the spatial distribution of the pixels in the image to be classified with size $L \times M$. Hence, we have one *FCM* network associated to each cluster. Each concept C_i in the net_j is associated to the pixel location (x,y) in the image, i.e. $i \equiv (x, y)$. Hence, the number of concepts in the net_j is $q = L \times M$ and a concept represents a pixel in the image. For simplicity, instead of using the sample $\boldsymbol{x}_s$ with its R, G, B spectral components at (x,y), we will say that the concept C_i is to be classified as belonging to a cluster w_j. The activation level for the concept C_i in the net_j is initialized with the membership degree μ_i^j provided by *FC* according to the equation (5), but mapped linearly for ranging in $[-1,+1]$ instead of $[0,+1]$. Hence, the activation levels start with the membership degrees associated to the concepts. Through the *FCM*, the activation levels are reinforced or punished iteratively based on the influences exerted by their neighbours. The goal is to make the best decision for classifying the concept C_i, based on more stable activation levels. Because we have one *FCM* for each cluster w_j, both the activation levels and causal edges are identified with the super-index j associated to the cluster w_j, i.e. according to the equation (1), now $A_i(t) \equiv \mu_i^j(t)$ and $e_{ik}^j(t)$ include j, this is applicable to the decay factor, identified as d_i^j. In summary, given an image of size $L \times M$, the goal is to classify the pixel i, equivalently the concept C_i, located at (x,y) as belonging to a cluster w_j The initial activation levels at $t = 0$ are set as $\mu_i^j(0) \equiv \mu_i^j$, computed through the equation (5). Every concept is positively or negatively activated to a certain degree. At the end of the iterative process, the decision about the cluster to which it belongs is made based on maximum activation levels considering all j networks.

Each causal weight $e_{ik}^j(t)$ is defined as a combination of two coefficients representing the mutual influence exerted by the k neighbours concepts C_k, over the concept C_i, they are:

a) a *regularization* coefficient which computes the consistency between the activation levels of the concepts and the supports provided by the classifier *BP* in a given neighbourhood for each net_j;

b) a *contextual* coefficient which computes the consistency between the clustering labels after each decision made by the combined classifier.

Considering the equation (1), the goal is to compute: *a*) the causal weights $e_{ik}^{j}(t)$ and *b*) the decay factor.

3.2.2 Computing the Causal Weights

Taking into account the mapping between a pixel location $i \equiv (x, y)$ and the concept C_i at each net_j, the neighbourhood $\boldsymbol{N}_i^m$ contains the m pixels (concepts) surrounding i, mapped from the image and representing the m-connected spatial region around the pixel i. A typical value of m used in the literature is 8, which defines a 3×3 region. As can be observed, the index i varies from 1 to q, i.e., this is the number of concepts explored at each net_j. For borders concepts in the image, the neighbourhood only includes the pixels belonging to the image, i.e., $m = 3$ in the four corners and $m = 5$ in the remainder borders.

We define the regularization coefficient at the iteration t as follows,

$$r_{ik}^{j}(t) = \begin{cases} 1 - \left| \mu_i^j(t) - p_k^j \right| & k \in \boldsymbol{N}_i^m, \quad i \neq k \\ 0 & k \notin \boldsymbol{N}_i^m, \quad i = k \end{cases} \tag{9}$$

where p_k^j is supplied by *BP*, it is the probability that a concept C_k (pixel k) with attributes $\boldsymbol{x}_k$ belongs to the cluster w_j, computed through the equation (7), i.e. $p_k^j \equiv p\left(\boldsymbol{x}_k \mid w_j\right)$. These values are mapped linearly to range between [−1, +1] instead of [0,+1]. From (9) we can see that $r_{ik}^{j}(t)$ ranges between [−1, +1] where the lower/higher limit means minimum/maximum influence respectively. The contextual coefficient for the concept C_i associated to the cluster w_j at the iteration t is defined taking into account the clustering labels l_i and l_k as follows,

$$c_{ik}(t) = \begin{cases} +1 & l_i(t) = l_k(t) \quad k \in \boldsymbol{N}_i^m, \quad i \neq k \\ -1 & l_i(t) \neq l_k(t) \quad k \in \boldsymbol{N}_i^m, \quad i \neq k \\ 0 & k \notin \boldsymbol{N}_i^m, \quad i = k \end{cases} \tag{10}$$

where values of −1 and +1 mean negative and positive influences, respectively.

Labels l_i and l_k could change between iterations and they are obtained as follows: given the concept C_i, we know its activation level, at each iteration t for each net_j as given by the equation (1); we determine that the concept C_i belongs to the cluster w_j if $\mu_i^j(t) > \mu_i^h(t) \ \forall j \neq h$, so we set $l_i(t)$ to the j value which identifies the cluster, j = 1,..., c. The label $l_k(t)$ is set similarly. Thus, this coefficient is independent of the net_j, because it is identical for all networks.

The regularization and contextual coefficients are both combined for computing the causal edges e_{ik}^{j} as an averaged sum, taking into account the signs, as follows,

$$z_{ik}^{j}(t) = \gamma r_{ik}^{j}(t) + (1-\gamma) c_{ik}(t); \quad e_{ik}^{j} = \left[sgn\left(z_{ik}^{j}(t) \right) \right]^{v} z_{ik}^{j}(t);$$
$$sgn\left(z_{ik}^{j}(t) \right) = \begin{cases} -1 & z_{ik}^{j}(t) \le 0 \\ +1 & z_{ik}^{j}(t) > 0 \end{cases} \tag{11}$$

$\gamma \in [0,1]$ represents the trade-off between both coefficients, a typical value for it is 0.8; *sgn* is the *signum function* and *v* is the number of negative values in the set $B \equiv \left\{ W_{ik}^{j}(t), r_{ik}^{j}(t), c_{ik}(t) \right\}$, i.e., given $D \equiv \left\{ u \in B / u < 0 \right\} \subseteq B$, *v* = card (*D*). Note that the causal edges vary with the iteration *t*. The Gestalt's similarity principle is mapped in $r_{ik}^{j}(t)$ through similarities between activation levels and probabilities and also in $c_{ik}(t)$ trough similarity between the labels. The other two Gestalt's principles of proximity and connectedness are embedded in the neighbourhood $\boldsymbol{N}_i^m$.

3.2.3 Computing the Decay Factor

We define the decay factor based on the assumption that high stability in the network states implies that the activation level for the concept C_i in the network net_j would be to lose some of its activation with such purpose. We build an accumulator of cells of size $q = L \times M$, where each cell *i* is associated to the concept C_i. Each cell *i* contains the number of times h_i^j, that the concept C_i has changed significantly its activation level in the net_j. Initially, all h_i^j values are set to zero and then $h_i^j = h_i^j + 1$ if $\left| \mu_i^j(t+1) - \mu_i^j(t) \right| > \varepsilon$. The stability of the node *i* is measured as the fraction of changes accumulated by the cell *i* compared with the changes in its neighbourhood $k \in \boldsymbol{N}_i^m$ and the number of iterations *t*. The decay factor is computed as follows,

$$d_i^j = \begin{cases} 0 & h_i^j = 0 \text{ and } h_k^j = 0 \\ \dfrac{h_i^j}{\left(\bar{h}_k^j + h_i^j \right) t} & \text{otherwise} \end{cases} \tag{12}$$

where h_i^j is defined above and $\bar{h}_k^j$ is the average value accumulated by the concepts $k \in \boldsymbol{N}_i^m$. As one can see, from equation (12), if $h_i^j = 0$ and $h_k^j = 0$, the decay factor takes the null value, this means that no changes occur in the activation levels of the concepts, i.e., high stability is achieved; if the fraction of changes is small, the stability of the node *i* is also high and the decay term tends towards zero. Even if the fraction is constant the decay term also tends to zero as *t* increases, this means that perhaps initially some changes can occur and then no

more changes are detected, and this is another sign of stability. The decay factor subtracts from the new activation level a fraction; this implies that the activation level could take values less/greater than $\tilde{1}$ or +1. In these cases, the activation level is set to $\tilde{1}$ or +1, respectively.

3.2.4 Summary of the Full FCM Process

In section 2, we introduced three criteria for convergence. So, based on them, the iterative process ends if all nodes in the network fulfil the convergence criterion $\left|\mu_i^j(t-1)-\mu_i^j(t)\right| > \varepsilon$ or a number of iterations t_{max}, is reached or a percentage of concepts, greater than a threshold U, change its activation level in a value greater than ε between two consecutive iterations. This last implies that the system is under a chaotic behaviour.

The FCM process is synthesized as follows:

1. *Initialization*: load each concept with its activation level $\mu_i^j(t=0) \equiv \mu_i^j$ through the equation (5); set $\varepsilon = 0.05$, t_{max} = 50 and U = 0.9. Define *nc* as the number of concepts from a total of q that change their activation levels at each iteration. The activation mechanism is that defined in equation (1) and the activation function is that defined in equation (4).

2. *FCM process*:
   $t = 0$
   while $t < t_{\max}$ *and* $nc/q < U$
   $t = t + 1$; $nc = 0$;
   for each concept C_i
   update $\mu_i^j(t)$ according to the equation (1)
   if $\left|\mu_i^j(t)-\mu_i^j(t-1)\right| > \varepsilon$ *then*
   $nc = nc + 1$
   end if;
   end for;
   end while

3. *Outputs*: the activation levels $\mu_i^j(t)$ for all concepts updated.

Once the FCMs processes end, each concept C_i has achieved an activation level $\mu_i^j(t)$ that determines de degree of belonging of the pixel *i*, represented by C_i, to the cluster *j*. This decision is made according to the following rule $i \in w_j$ if $\mu_i^j(t) > \mu_i^h(t)$, $\forall\, j \neq h$ where *j* and *h* identify the clusters and *t* represents the last iteration.

Figure 2(*a*) displays an original image, which is to be classified and (*b*) the image classified after the FCM process is applied with eighteen iterations.


(*a*)

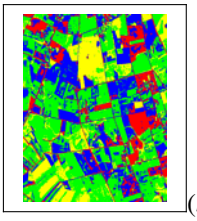

(*b*)

Fig. 2 Original image; (*b*) classification of the pixels in four clusters through the FCM process

4 Application to Image Change Detection

The image change detection is the second problem addressed under the FCM framework in this work. This problem is formulated as follows: given two registered images $I_1(x,y)$, $I_2(x,y)$ of size $L \times M$ of the same area in the scene, taken at different times, the goal is to detect if a pixel, located at (x,y), has changed and the magnitude of the change. The approach proposed here is based on the work of Pajares et al. (2007) and Pajares (2006).

A difference image D, is computed at each pixel location (x,y) by subtracting the corresponding intensity values of the incoming images, i.e. $D(x,y) = I_2(x,y) - I_1(x,y)$. We also formulate two hypotheses H_0 and H_1 that represent no change and change respectively.

Bruzzone and Fernandez (2000) proposed an image change detection technique that estimates the parameters of the mixture distribution $p(D)$ consisting of all pixels in the difference image. The mixture distribution $p(D)$ can be written as,

$$p\left(D(x,y)\right) = p\left(D(x,y)\middle|H_0\right)P\left(H_0\right) + p\left(D(x,y)\middle|H_1\right)P\left(H_1\right) \tag{13}$$

Under this assumption, the probability density functions $p\left(D(x,y)\middle|H_0\right)$, $p\left(D(x,y)\middle|H_1\right)$ and the *a priori* probabilities $P\left(H_0\right)$ and $P\left(H_1\right)$ are estimated by using the Expectation Maximization (*EM*) algorithm (Duda et al., 2000), which is a general approach to maximum-likelihood estimation. It is assumed that both $p\left(D(x,y)\middle|H_0\right)$ and $p\left(D(x,y)\middle|H_1\right)$ can be modeled by Gaussian distributions. So, the parameters to be estimated are the means μ_0, μ_1 and variances σ_0^2, σ_1^2 respectively. The process is iterated until convergence and the initial values of the estimates are determined by simple differencing. Based on these estimates, we can determine if a pixel location can be classified as changed or unchanged. Nevertheless, better decisions are still possible by applying the Gestalt's principles that consider the spatial information existing between a pixel and its neighbors. This improvement is achieved through the FCMs.

Under the FCM framework, we build a network of $q = L \times M$ concepts, where each concept C_i represents a pixel i at the image location (x,y). The activation level of each concept at each iteration $A_i(t)$ determines the magnitude of the

change at this location, ranging from $A_i(t) = +1$, maximum degree of change to $A_i(t) = -1$, without change.

The initialization of the activation levels is carried out by exploiting the characteristics of the difference image. We use the initialization strategy, described in Bruzzone and Fernandez (2000), as follows, from the histogram of the difference image $h(D)$ we compute two different thresholds T_0 and T_1 as $T_0 = M_D(1-\alpha)$ and $T_1 = M_D(1+\alpha)$, where M_D is the middle value of $h(D)$, i.e. $M_D = \left[\max\{D\} - \min\{D\}\right]/2$, and $\alpha \in (0,1)$ is set to 0.5 in this process. Now, given a pixel location (x,y) in the difference image, the initial activation level of its associated concept is given by equation (14).

$$A_i(0) = \begin{cases} -1 & if \quad D(x,y) < T_0 \\ +1 & if \quad D(x,y) > T_1 \\ -1 \text{ or } +1 \ (randomly) & \text{otherwise} \end{cases} \tag{14}$$

Once we have initialized the network, the next step consists in the definition of the edges $e_{ik}(t)$ based on the application of the Gestalt's principles. This is carried out by computing the data and contextual consistency coefficients. Afterwards, the decay factor is also derived. This allows updating the initial activation levels according to the equation (1).

4.1 Computation of the Causal Weights: Data and Contextual Consistencies

The data information is mapped through the *a posteriori* probabilities that given a pixel value $D(x,y)$ in the difference image D it is associated to hypothesis $H_k \in \{H_0, H_1\}$, i.e. we compute $P(H_k | D(x,y))$. This is carried out by applying the Bayes rule,

$$P(H_k | D(x,y)) = \frac{p(D(x,y)|H_k)P(H_k)}{p(D(x,y))} \tag{15}$$

where the *density functions* $p(D(x,y)|H_k)$ and the *a priori* probabilities $P(H_k)$, are estimated through the EM algorithm as described above. The mixture density distribution $p(D(x,y))$ is computed trough equation (13). The initialization required by the EM algorithm is based on the procedure described above, under the assumption that $D(x,y)$ is with hypothesis H_0 or H_1 if $D(x,y) < T_0$ or $D(x,y) > T_1$ respectively. As we have not prior information, the *a priori* probabilities are initialized to 0.5.

Now, each pixel (x,y) in the difference image $D(x,y)$ should be associated to the hypothesis that maximizes the posterior conditional probability, i.e.

$$\begin{aligned} H_s &= \arg \max_{H_s \in \{H_0, H_1\}} \left\{ P\left(H_s \middle| D(x,y)\right) \right\} \\ &= \arg \max_{H_s \in \{H_0, H_1\}} \left\{ P(H_s)\, p\left(D(x,y) \middle| H_s\right) \right\} \end{aligned} \tag{16}$$

From (16) we build a *data map* with the same size as the difference image and identical (x,y) locations that those of the pixels in the image difference and concepts in the FCM network. Each concept C_i associated to the pixel location $i \equiv (x, y)$ is loaded with the data information $r(i)$ according to the criterion in equation (16) as follows,

$$a(i) = (-1)^{s+1} P\left(H_s \middle| D(x,y)\right); \quad s = \{0,1\} \tag{17}$$

When a pixel i is considered unchanged, it is with the hypothesis H_0 and $a(i) = -P\left(H_0 \middle| D(x,y)\right)$ with its minimum value being -1. On the contrary, if a pixel has changed significantly it is with the hypothesis H_1 and $a(i) = P\left(H_1 \middle| D(x,y)\right)$ with its maximum value being +1.

Now, the goal is to map the data consistency between concepts C_i and C_j into the consistency coefficient r_{ij}. Given the concept C_i we consider its m-connected neighborhood, $\boldsymbol{N}_i^m$, where m takes the typical value of 8.

For each concept C_i, only consistencies can be established between concepts C_j, where the associated pixels $j \in \boldsymbol{N}_i^m$ and $i \neq j$ otherwise if $k \notin \boldsymbol{N}_i^m$ it is assumed that there is not consistency between concepts C_i and C_k. This is justified under the hypothesis that only local relations can be established between changed/unchanged pixels, based on the Gestalt's proximity principle. Indeed, given a changed pixel probably its neighbors should be also changed pixels and vice versa for unchanged pixels. Two concepts C_i and C_j where $j \in \boldsymbol{N}_i^m$ are said consistent if they have similar data information. Otherwise they should be inconsistent.

The data consistency between the concepts C_i and C_j is mapped into the coefficient r_{ij} a*s* follows,

$$r_{ij} = \begin{cases} 1 - \left| a(i) - a(j) \right| & j \in \boldsymbol{N}_i^m \\ 0 & j \notin \boldsymbol{N}_i^m \end{cases} \tag{18}$$

r_{ij} ranges in $\tilde{[}1,+1]$, where $-1/+1$ represents strong unchanged/changed respectively; the contribution so made may be positive (excitatory causality) or negative (inhibitory causality). Hence, a positive data consistency will contribute towards the network stability, in terms of small variations on its activation levels.

The mapping of the contextual consistencies involves the current activation levels according to the equation (19).

$$c_{ij}(t) = \begin{cases} h_{ij}\left(1 - \left| A_i(t) - A_j(t) \right|\right) & j \in \boldsymbol{N}_i^m \\ 0 & j \notin \boldsymbol{N}_i^m \end{cases} \tag{19}$$

where h_{ij} is a coefficient that measures the influence exerted by the concept C_i over the concept C_j taking into account the relevance of each concept. The relevance is a magnitude introduced in this approach in order to measure the strength of each concept against changes on its activation level. We build an accumulator of cells of size $q = L \times M$, where the cell i is associated to the concept C_i. Each cell i contains the number of times, h_i, that the concept C_i has changed significantly its activation level. Initially, all h_i are set to zero and then $h_i = h_i + 1$ if $\left|A_i(t+1) - A_i(t)\right| > \varepsilon$, where ε is set to 0.05 as in the classification approach; h_{ij} is computed as follows,

$$h_{ij} = \begin{cases} h_j \big/ \left(h_i + h_j\right) & \left(h_i + h_j\right) \geq 2 \\ 1 & \text{else} \end{cases} \tag{20}$$

The equation (20) measures the fraction of changes accumulated for the concept C_j as compared to the concept C_i; h_{ji} measures the reverse influence of concept C_j over concept C_i. This equation is interpreted as follows, if $h_i < h_j$ the concept C_i has accumulated less number of changes than the concept C_j, i.e. the relevance of C_i is greater than the concept C_j and vice versa. This implies that h_{ij} could be different from h_{ji}. From (19) we can see that c_{ij} varies with the iteration and ranges in [−1,+1] where the lower/higher limit means minimum/maximum contextual consistency respectively. One can see that c_{ij} could be different from c_{ji} (non symmetry) due to the factors h_{ij} or h_{ji}.

Now the goal is to combine appropriately r_{ij} and $c_{ij}(t)$ in order to derive the causal weights required by the equation (1). Making use of the fuzzy set theory, we consider two fuzzy sets, where their elements are pairs of concepts (C_i, C_j) and the degrees of compatibility (membership functions) are given by r_{ij} and $c_{ij}(t)$ respectively. According to the dissertations of Zimmermann (1991) we propose the Hamacher's union operator for the combination because of its performance,

$$e_{ij}(t) = \frac{\left(\gamma - 1\right) r_{ij} c_{ij}(t) + r_{ij} + c_{ij}(t)}{1 + \gamma c_{ij}(t) r_{ij}} \tag{21}$$

where $\gamma \geq -1$; by setting $\gamma = -1$ the Hamacher's union operator matches with the Hamacher sum. We have used this value in the proposed approach.

The Gestalt's principle of similarity is considered in both r_{ij} and $c_{ij}(t)$ coefficients through the differences in the data and activation levels respectively. As in the above classification approach the proximity and connectedness principles are embedded in the neighborhood N_i^m.

4.2 Computing the Decay Factor

The decay factor involved in the equation (1) is computed in a similar fashion that the one defined in the equation (12). Indeed, considering the accumulated changes for each concept and its neighbors, we can write the equation (22)

$$d_i = \begin{cases} 0 & h_i = 0 \text{ and } \overline{h_k} = 0 \\ \dfrac{h_i}{\left(\overline{h_k} + h_i\right)\, t} & \text{otherwise} \end{cases} \tag{22}$$

where h_i is defined in section 4.1 and $\overline{h_k}$ is the average value accumulated by the concepts $k \in N_i^m$. The same discussion with respect the equation (12) can be applied here.

4.3 Summary of the Full FCM Process

The FCM process is synthesized as follows:

1. *Initialization*: load each concept with its activation level $A_i(t=0)$ through the equation (14); set $\varepsilon = 0.05$, t_{max} = 50 and U = 0.9, which are the typical values already used in the classification procedure. Define *nc* as the number of concepts from a total of *q*, that change their activation levels at each iteration. The activation mechanism and the activation function are defined in (1) and (4) respectively.

2. *FCM process*:
   t = 0
   while $t < t_{max}$ *and* $nc/q < U$
   $t = t + 1$; $nc = 0$;
   for each concept C_i
   update $A_i(t)$ according to the equation (1)
   if $|A_i(t) - A_i(t-1)| > \varepsilon$ *then*
   $nc = nc + 1$
   end if;
   end for;
   end while

3. *Outputs*: the activation levels $A_i(t)$ for all concepts updated.

Once the FCMs processes end, each concept C_i has achieved an activation level $A_i(t)$ that determines de degree of change. So, we consider that a pixel has changed if $A_i(t) > \varepsilon$, where its value in the range $[-1,+1]$ determines the degree.

Figure 3(*a*) and (*b*) display two original images from an outdoor environment, where the changes between them are to be identified. After the FCM process, the changes detected are displayed in image 2(*c*), and represented in the gray scale from 0 to 255, so that 255 means no change and 0 maximum degree of change. Intermediate values represent the magnitude of the change.

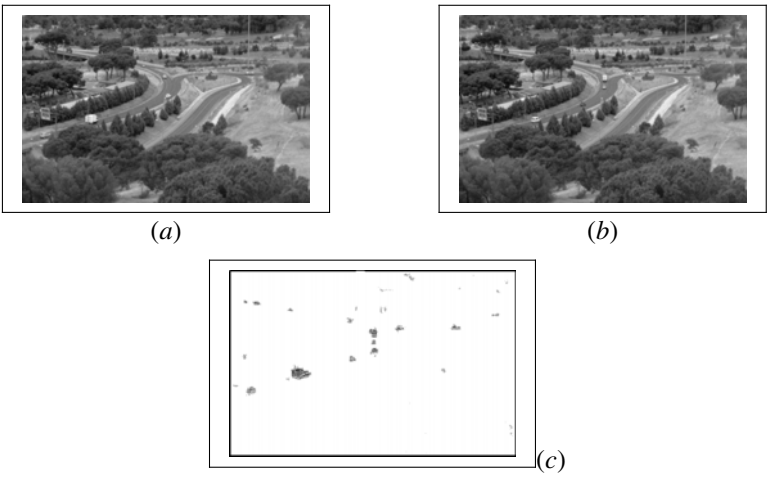


Fig. 3 Outdoor environment; (*a*) and (*b*) two images of the same sequence and scene; (*c*) changes detected with the FCM approach

5 Application to Image Matching in Stereovision

This is the third application addressed in this work. The image matching in stereovision is the process of identifying the corresponding points in two images that are cast by the same physical point in the tri-dimensional space. This can be carried out pixel by pixel or identifying significant features in the images, such as edges, regions or interest points. This is the approach used in this section, where we try to match pairs of edge segments.

Hence, the stereo correspondence problem can be defined in terms of finding pairs of true matches, namely, pairs of edge segments in two images that are generated by the same physical edge segment in space. These true matches generally satisfy some constraints (Scharstein and Szeliski, 2002): 1) *epipolar*, given two segments one in the left image and a second in the right one, if we slide one of them along a horizontal direction, i.e. parallel to the epipolar line, they would intersect (overlap) (figure 4); 2) *similarity*, matched edge segments have similar properties or attributes; 3) *smoothness*, disparity values in a given neighbourhood change smoothly, except at a few depth discontinuities; 4) *ordering*, the relative position among two edge-segments in the left image is preserved in the right one for the corresponding matches; 5) *uniqueness*, each edge-segment in one image should be matched to a unique edge-segment in the other image.

In section 5.1, we define the procedure for extracting edge segments and computing the attributes of each one of them. Afterwards, we build the FCM structure, where each pair of edge segments will define a concept in the FCM. The similarity constraint will allow define the activation levels of the concepts (section 5.2) and the smoothness, ordering and epipolar constraints will allow define the causal edges (section 5.3). The approach described in this chapter is based on the work of Pajares and Cruz (2006).

5.1 Feature and Attribute Extraction

The contour edges in both images are extracted using the Laplacian of Gaussian filter in accordance with the zero-crossing criterion Huertas and Medioni (1986). For each zero-crossing in a given image, its gradient vector (magnitude and direction) Leu and Yau (1991) and Lew et al. (1994) and variance Krotkov (1989) values are computed from the gray levels of a central pixel and its eight immediate neighbors. The edges are obtained by joining adjacent zero-crossings following the algorithm of Tanaka and Kak (1990), where: (1) a margin of deviation of ± 20% in gradient magnitude and of ±45° in gradient direction are allowed; (2) each detected contour is approximated by a series of piecewise linear line segments (Nevatia and Babu, 1980). Finally, for every segment, an average value of the four attributes is obtained from all computed values of its zero-crossings. All average attribute values are normalized in the same range. Now, we have edge segments, which are the features and their attributes.

Each pair of features has two associated 4-dimensional vectors $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$, where the components are the attribute values, and the sub-indices a and b denote features belonging to the left and right images, respectively. A four-dimensional difference measurement vector $\boldsymbol{x}$ is then also obtained from the above $\boldsymbol{x}_a$ and $\boldsymbol{x}_b$ vectors, $\boldsymbol{x} = \boldsymbol{x}_a - \boldsymbol{x}_b = \{x_m, x_d, x_l, x_v\}$. The components of $\boldsymbol{x}$ are the corresponding differences for module and direction gradient, Laplacian and variance values. Only those pairs verifying the following three initial conditions will be processed: 1) their absolute value of the difference in the gradient direction is below a specific threshold, fixed to 25º in this work; 2) their absolute value in the gradient magnitude is also below a fixed threshold, set to 15, and 3) their overlap rate surpasses a certain value, fixed to 0.5 here. The remaining pairs that do not meet such conditions are directly considered as false correspondences. The overlap is a concept introduced in Medioni and Nevatia (1985), two segments u and z overlap if by sliding one of them in a direction parallel to the epipolar line, they would intersect.

Figure 4 clarifies the overlapping concept. Indeed, segment u in the left image overlaps with segment s in the right image, but segment v does not overlap with s. The overlap rate between edge segments (u,z), α_{uz} is defined as the percentage of coincidence, ranging in [0,1], when two segments u and z overlap, and it is computed taking into account the common overlap length l_c defined by c and the two lengths for the involved edge segments l_u and l_z respectively. All lengths are measured in pixels.

$$\alpha_{uz} = 2l_c / (l_u + l_z) \tag{23}$$

Considering the above initial conditions 1) and 3) and the parallel optical axis geometry of the stereovision system, we compute the disparity between two edge-segments (u and z in figure 1) as follows: trace epipolar lines (four) crossing the common overlapping segment (c), for each line compute x_u and x_z, so the disparity is $(x_u - x_z)$. Then the disparity for both edge segments u and z is the averaged disparity for the four pairs of points x_u and x_z.

Figure 4 also illustrates the neighboring N_{ab} concept between edge-segments (a,b). For each edge segment "a" in the left image we define a window $w(a)$ in the right image and, similarly, for each segment "b" in the right image, we define a window $w(b)$ in the left image. Given the pair (a,b), a new pair (g,o) belongs to N_{ab}, if g lies in $w(b)$ and o lies in $w(a)$. It is said that "a segment g lies in $w(b)$" if at least the 30% of the length of the segment "g" is contained in the $w(b)$ window. The shape of this window is a parallelogram; one side is "a", for left to right match, and the other a horizontal vector of length $2.maxd$. The constant $maxd$ is said the disparity limit, it is set to 15 pixels in the proposed approach and is used for mapping the smoothness constraint as an interconnection value between concepts.

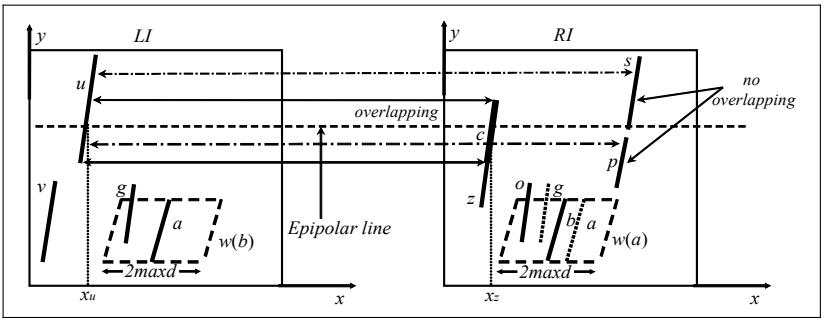


Fig. 4 Overlapping concept, edge-segments interactions and neighborhood for the pair (a, b)

5.2 Causal Concepts and Their Activation Levels

The system receives as inputs a pair of stereo images left (*LI*) and right (*RI*). This pair is processed in order to extract *edge segments* and their attributes; each pair of extracted features (a,b) is to be matched, the features a and b come from *LI* and *RI* respectively. For each pair (a,b) the attribute difference vector $\boldsymbol{x}$ is computed as described above. In this approach, a pair of edge segments (a,b) defines a causal fuzzy concept C_i, where its initial activation level at the iteration $t = 0$ is derived from $\boldsymbol{x}$ as follows,

$$A_i(0) = \left(1 + \|\boldsymbol{x}\|\right)^{-1} \tag{24}$$

where $\|\cdot\|$ is defined as the Euclidean norm, i.e. A_i ranges from 0 to +1 if $\|\boldsymbol{x}\| \to \infty$ (minimum similarity between a and b) or $\|\boldsymbol{x}\| = 0$ (maximum similarity between a and b respectively. This implies the application of the similarity Gestalt's principle. Hence, our FCM structure is built with as many concepts as pairs of edge segments, from *LI* and *RI*, are available.

5.3 Computation of the Causal Weights

Based on the smoothness, ordering and epipolar stereovision matching constraints, we map them for computing the causal weights between concepts. This is carried out in the next three subsections. Afterwards, we describe the *FCM* process, after which the uniqueness constraint is applied.

5.3.1 Mapping the Smoothness Constraint

The smoothness constraint assumes that neighbouring edge segments have similar disparities, except at a few depth discontinuities (Medioni and Nevatia, 1985). This implies the mapping of the proximity and connectedness Gestalt's principles. Generally, when the smoothness constraint is applied, it is assumed there is a bound on the disparity range allowed for any given segment. This limit is denoted as *maxd* and defined before.

Returning to figure 4, given "*a*" and "*g*" in *w*(*b*) and "*b*" and "*o*" in *w*(*a*) where "*a*" matches with "*b*" and "*g*" with "*o*" the differential disparity $|d_{ab} - d_{go}|$, measures how close the disparity between edge segments "*a*" and "*b*" denoted as d_{ab} is to the disparity d_{go} between edge segments "*g*" and "*o*". The disparity between edge segments is the average of the disparity between the two edge segments along the length they overlap. This differential disparity criterion is used in Ruichek and Postaire (1996), Medioni and Nevatia, (1986), Pajares and Cruz (2004, 2006) among others. We define a compatibility coefficient derived from Nasrabadi and Choo (1992) and Ruichek and Postaire (1996) as follows,

$$c_{(ab)(go)}(D) = \frac{1}{1+\exp\left[\gamma\left(D/m(D)-1\right)\right]} \tag{25}$$

where $D = \left|d_{ab} - d_{go}\right|$, *m*(*D*) denotes the average of all values *D* in the pair of stereo images (*LI* and *LR*) under processing. The slope of the compatibility coefficient in (25) is expressed by γ and varies for each pair of stereo images. To determine γ, it is assumed that the probability distribution function of *D* is Gaussian with average *m*(*D*) and standard deviation $\sigma(D)$, i.e. $p(D) = \left[1+\exp\left[\gamma\left(D_{(ab)(go)}/m(D)-1\right)\right]\right]^{-1}$. Under this assumption and following Barnard (1989) and Hattori (1998), to set the possibility value to 0.1 when the value of cumulative distribution function is 0.9, γ value is calculated by $\gamma = \ln 9\left(\left(m(D)\right)/\left(1.282\sigma(D)\right)\right)$. In this approach, typical values of γ, *m*(*D*) and $\sigma(D)$ are about 6, 9 and 2 respectively. So, values of *D* near 0 should give high values in the compatibility coefficient $c_{(ab)(go)}(\cdot) \approx +1$, but near 25 they give low values, $c_{(ab)(go)}(\cdot) \approx 0$. Note that $c_{(ab)(go)}(\cdot)$ ranges in (0,1). So, a compatibility coefficient of +1 is obtained for a good consistency between nodes (*a*,*b*) and (*g*,*o*) (*D* = 0) and a compatibility of 0 for a bad consistency ($D >> 0$). This causal weight embedding the smoothness constraint should indicate positive causality for high compatibility coefficient values and vice versa.

5.3.2 Mapping the Ordering Constraint

We define the ordering coefficient $O_{(ab)(go)}$, for the edge-segments according to (26), which measures the relative average position of edge segments *"a"* and *"g"* in *LI* with respect to *"b"* and *"o"* in *RI*, related to the neighboring N_{ab}, it ranges from 0 to 1.

$$O_{(ab)(go)} = -\frac{1}{S}\sum_{S} y_{(ab)(go)}$$
$$\text{where} \quad y_{(ab)(go)} = \left| R(x_a - x_g) - R(x_b - x_o) \right| - 1 \quad \text{and} \quad R(r) = \begin{cases} 1 & \text{if } r>0 \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

We trace S scanlines (four) along the common overlapping length, each scanline produces a set of four intersection points (a_S and g_S in *LI* and b_S and o_S in the *RI*) with the four edge-segments. Hence, the $y_{(ij)(hk)}$ can be computed as in Ruichek and Postaire (1996) considering the above four edge points and it takes 0 and 1 as two discrete values. A value of +1 in the ordering coefficient means that the ordering constraint is preserved. On the contrary, a value of 0 indicates that the ordering constraint is not preserved. The causal weight embedding the ordering constraint should indicate positive causality for a high ordering coefficient value.

5.3.3 Mapping the Epipolar Constraint (Overlapping Concept)

The epipolar constraint is mapped through the overlapping concept described in Medioni and Nevatia (1985), by computing the overlapping coefficient as given in equation (27).

$$\lambda_{(ab)(go)} = 0.5\left(\alpha_{ab} + \alpha_{go}\right) \tag{27}$$

where α is the overlap rate defined in equation (23). Under the epipolar constraint we can assume that correct/incorrect matches should have high/low overlap rates, i.e. the overlapping coefficient should be +1 or 0 respectively and $\lambda_{(ab)(go)}$ for neighborhoods should be high increasing the consistency. The use of the overlapping criterion is justified by the fact that the edge segments are reconstructed by piecewise linear line segments as described in section 5.1. The reasoning for the influence of this coefficient in the causal weight is similar to the previous ones for the compatibility and ordering coefficients.

5.3.4 Fuzzy Criteria for Computing the Causal Weights

Once the smoothness, ordering and epipolar stereovision matching constraints have been mapped, we have available the compatibility, ordering and overlapping coefficients derived from such constraints respectively. Now the goal is to combine the three coefficients for computing the causal weights involved in equation (1). Making use of the fuzzy theory, we can consider these coefficients as membership functions, which can be combined in order to compute the causal weight $e_{(ab)(go)}$.

Hence, in our approach the causal weight is considered as a fuzzy measurement (membership value). Taking into account the dissertations in Zimmermann (1991), a straightforward approach for aggregating fuzzy sets, would be to use the aggregating procedures frequently used in multi-criteria decision theory. They realize the idea of trade-offs between conflicting goals when compensation is allowed, and the resulting trade-offs lie between the most optimistic lower bound and the most pessimistic upper bound, that is, they map between the minimum and the maximum degree of membership of the aggregated sets. Therefore they are called averaging operators.

Following the discussion in Zimmermann (1991) about the criteria for selecting appropriate aggregation operators, we find that adaptability is suitable; this can be achieved by parametrization. Thus *min* and *max* operators cannot be adapted at all. They are acceptable in situations in which they fit, by contrast, there are other operators hat can be adapted to certain contexts by setting their parameters; we have used the Hamacher's union operator. Taking into account that causal weights are considered as fuzzy membership values and making use of the operator's associativity the equation (28) is derived. The parameter τ allows a fitting appropriately

$$\begin{aligned}\rho_{(ab)(go)} &= \frac{(\tau-1)c_{(ab)(go)}O_{(ab)(go)} + c_{(ab)(go)} + O_{(ab)(go)}}{1+\tau c_{(ab)(go)}O_{(ab)(go)}} \\ E_{(ab)(go)} &= \frac{(\tau-1)\rho_{(ab)(go)}\lambda_{(ab)(go)} + \rho_{(ab)(go)} + \lambda_{(ab)(go)}}{1+\tau\rho_{(ab)(go)}\lambda_{(ab)(go)}}\end{aligned} \tag{28}$$

where $\tau \geq -1$, we have found acceptable the behavior of this parameter by setting it to 1. This is because its behavior is a trade-off between maximum and minimum operators Zimmermann (1991). Other values for τ and other parameterized operators (Einstein, Yager, Dubois and Prade among others, Zimmerman) could be used.

As a result of the aggregation's operators, the resulting $E_{(ab)(go)}$ from equation (28) ranges in [0,+1]. So, rescaling this interval to [−1,+1], we can derive the final causal weight $e_{(ab)(go)}$ between features (a,b) and (g,o) as required by the general equation (1). Hence, we obtain,

$$e_{(ab)(go)} = 2E_{(ab)(go)} - 1 \tag{29}$$

So, high coefficient values should give high causal weights i.e. positive causality as expected and vice versa for low coefficient values and negative causality.

5.3.5 Full FCM Process

The FCM process is synthesized as follows; it is similar to the processes defined in sections 3.2.5 and 4.3:

1 Initialization: load each concept with its activation level $A_i(t=0)$ through the equation (24); set $\varepsilon = 0.05$, t_{max} = 50 and U = 0.9, which are typical values used in the classification procedure, described in the section 3.2.5. Define *nc* as the number of concepts from a total of *q* concepts representing pairs of edge segments that change their activation levels at each iteration. The activation mechanism is that defined in equation (1), but with null decay factors and the activation function is that defined in equation (2).

2 FCM process:
$t = 0$
while $t < t_{\max}$ *and* $nc/q < U$
$t = t + 1$; $nc = 0$;
for each concept C_i
update $A_i(t)$ according to the equation (1)
if $|A_i(t) - A_i(t-1)| > \varepsilon$ *then*
$nc = nc + 1$
end if; end for; end while

3 Outputs: the activation levels $A_i(t)$ for all concepts updated.

5.3.6 Uniqueness Constraint

The *uniqueness* constraint completes the set of matching constraints used for solving the stereovision matching problem.

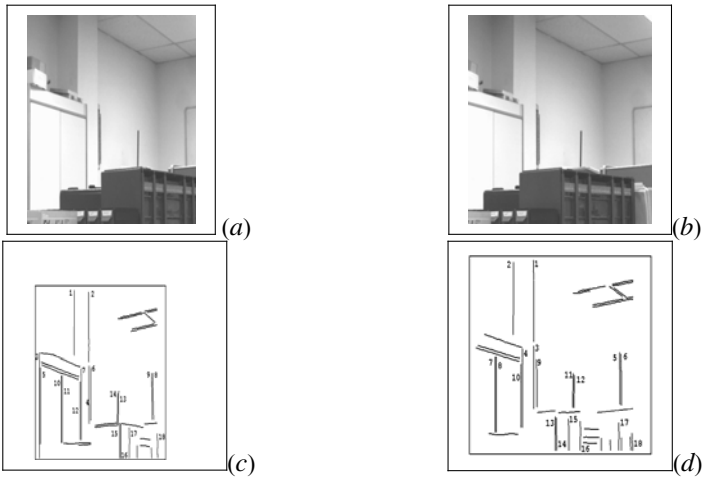


Fig. 5 (*a*) and (*b*) original left and right stereo images; (*c*) and (*d*) labelled edge segments in left and right images

A left edge segment can be assigned to a unique right edge segment (unambiguous pair) or several right edge segments (ambiguous pairs). The decision about whether a match is correct, is made by choosing the greater activation level for the corresponding causal concept (in the unambiguous case there is only one) whenever it surpasses a previous fixed threshold U (=0.5), intermediate value for C_i ranging in [0, 1]. A secure true match (a,b), represented by the causal concept C_i should have at the last iteration an activation level with value the unity.

6 Comparative Analysis and Performance Evaluation

This section contains the corresponding comparative analysis and performances of the proposed methods, as compared to other strategies of similar category. Details about these performances wit respect image classification, image change detection and stereovision matching are provided in sub-sections 6.1 to 6.3 respectively.

6.1 Performance of FCMs in Texture Classification

In texture classification there are two phases involved: training and decision. We used a set of 36 digital aerial images acquired during May, 2006, from the Abadin region of Lugo (Spain). They are multi-spectral images, 512×512 pixels in size. The images were taken during different days from an area with several natural textures. We selected randomly 12 images from the set of 36 available. This set was used for training and the remainder set of 24 images for testing. Each image in the training set is down-sampled by two, obtaining the training samples used in this initial training phase, i.e., the number of training samples was n_0 = 12×256×256 = 786432.

We assume known the number of clusters, which is fixed to four in our experiments because this is the number of the main textures identified; hence, from the training set we estimate the cluster centres, $\boldsymbol{v}_j$, through *FC*, according to equation (5) and also the mean centres $\boldsymbol{m}_j$, and the covariance matrices Σ_j based on *BP* through equation (8).

Our combined *FCM* (*FM*) method was compared with the base classifiers used for the combination, i.e. *FC* and *BP*. *FM* was also compared with the following classical hybrid strategies described in Kittler et al. (1998): Mean (*ME*), Max (*MA*) and Min (*MI*). Given the node i, with supports of belonging to the cluster j, μ_i^j and p_i^j provided by *FC* and *BP* respectively, *ME* computes the mean value, i.e., $\left(\mu_i^j + p_i^j\right)/2$; *MA* the maximum value, i.e., $\max\left\{\mu_i^j, p_i^j\right\}$ and *MI* the minimum value, i.e., $\min\left\{\mu_i^j, p_i^j\right\}$. They were studied in terms of reliability (Cabrera, 2006). Given the resulting combined supports that the pixel i belongs to each cluster j, the final decision is made based on the maximum combined support value for all clusters. Yager (1988) also proposed a multi-criteria decision-making approach based

on fuzzy sets aggregation. So, *FM* was also compared against the *Fuzzy Aggregation* (*FA*) defined by the equation (30).

$$\gamma_i^j = 1 - \min\left\{1, \left(\left(1-\mu_i^j\right)^a + \left(1-p_i^j\right)^a\right)^{1/a}\right\} \qquad a \geq 1 \tag{30}$$

This represents the joint support provided by *FC* and *BP*; the decision is made based on the maximum support for all clusters as above. The parameter *a* is adjusted from the set of 12 images available for training. The experiments were carried out by varying the parameter *a* from 1 to 8. The minimum error rate was achieved with *a* = 4, which in the end was the value used for this parameter. The decision made according to node *i* with the supports provided by (30) was based on the maximum value, in keeping with the following rule: $i \in w_j$ if $\gamma_i^j > \gamma_i^h \quad \forall h \neq j$.

To verify the performance for each method we have built a ground truth for each image under the supervision of the expert human criterion. Based on the above assumption that the number of clusters is four, we classify each image pixel with the simple classifiers, obtaining a labelled image with four expected clusters. For each cluster we build a binary image, which is manually touched up until a satisfactory classification is obtained under the human supervision. The pixels belonging to a cluster are labelled as white.

Table 1 shows the percentage of error during the classification for the different classifiers. These percentages are computed as follows. Let I_r an image from the testing set of 24 for testing, (r = 1,…,24); i is the node at the location (x,y) in I_r. An error counter E_r is initially set to zero for each image r and for each classifier. Based on the corresponding decision process, each classifier determines the cluster to which the node i belongs, $i \in w_j$. If the same pixel location on the corresponding ground truth image is black, then the pixel is incorrectly classified and the error counter value is increased by the unity. The error rate of the image I_r is: $e_r = E_r / Z$, where Z is the image size, i.e., 512×512. The average error rate for the testing set is finally computed by: $\bar{e} = \frac{1}{24}\sum_{r=1}^{24} e_r$. The standard deviation is simultaneously computed as: $\bar{\sigma} = \sqrt{\frac{1}{23}\sum_{r=1}^{24}\left(e_r - \bar{e}\right)^2}$.

In table 2, they are displayed as percentages, i.e., $\tilde{e} = 100\bar{e}$ and $\tilde{\sigma} = 100\bar{\sigma}$. The numbers in square brackets in the row *FM* indicate the rounded and averaged number of iterations required for this method.

From results in table 1, it is seen that the best performance is achieved by the proposed *FM* strategy. The best performance for the classical hybrid methods is achieved by *ME* and for the simple methods is *BP*. The best performances are established in terms of the least average percentage of error and the least standard deviation values.

Table 1 Average percentages of error and standard deviations

$\tilde{e}$: average percentage of error $\tilde{\sigma}$: standard deviation of error (%)		$\tilde{e}$	$\tilde{\sigma}$
Iterative and fuzzy hybrid methods	**[iterations]** **FM** (Fuzzy Cognitive Maps)	**[12]** 17.5	0.85
	FA(Fuzzy Aggregation)	20.6	1.55
Classical hybrid methods	**MA** (Maximum)	26.3	2.01
	MI (Minimum)	29.1	2.35
	ME (Mean)	24.2	1.7
Simple methods	**FC** (Fuzzy clustering)	25.9	2.0
	BP(Bayesian Parametric)	24.7	1.8

6.2 Performance of FCMs for Image Change Detection

For comparative purposes, in image change detection, we used four data sets: 1) 40 pairs of real video sequences of outdoor environments of size 1392 *x* 1040; 2) 36 pairs of real video sequences of indoor environments of size 840 *x* 760; 3) 10 pairs of real remote sensing images of size 400 *x* 400; and 4) 30 pairs of synthetic sensing images also of size 400 *x* 400. A full description of this data set can be found in Pajares (2006), where a Hopfield Neural Network (HNN) is applied.

The analysis requires a ground truth for each pair of images tested, useful to assess change-detection errors and successes. So, we prepare a map of changed and unchanged areas as follows. In Rosin and Ioannidis (2003) are evaluated several classical global image thresholding approaches for image change detection. Based on this study, the best performance is achieved with the method described in Kapur et al. (1985) which uses the entropy of the histogram. In Wu et al. (2005) also a new technique based on cumulative histograms is used with acceptable performance. By applying both methods we obtain two binary images. They are combined by using the logical operator "*or*". The resulting image is manually refined and the *ground truth map* obtained.

The HNN method of Pajares (2006) was compared against six existing image change detection strategies including the approach of Bruzzone and Fernández-Prieto (2000) (BRU). The best performances were obtained by HNN followed by BRU. Now, in the present work, we compare the performance of the proposed FCM approach against the iterative HNN and BRU methods. FCM, HNN and BRU use the same initialization process and also a neighborhood region of size 3*x*3. We use a set of seven experiments, a brief description of the experiments is the following: E1: 30 outdoor pairs of images from the same sequence; E2: 10 outdoor pairs of images from different sequences of the same scene; E3: 12 indoor pairs of images from the same sequence without changes in the illumination levels; E4: 12 indoor pairs of images from the same sequence, during the full video capture the illumination levels are on-line changed, i.e. the images have different intensity levels; E5: 12 indoor pairs of images from the same sequence, an image is obtained without changes in the illumination during its capture and the other, as before, by varying on-line the illumination; E6: 10 pairs of remote sensing images of the same scene; E7: 30 pairs

of synthetic remote sensing images of the same scene corrupted with Gaussian noise of zero-mean and different variances selected randomly.

The results obtained for each method are compared against the ground truth, based on the *PCC* magnitude described in Rosin and Ioannidis (2003), also used in Pajares (2006): $PCC = (TP+TN)/(TP+FP+TN+FN)$, where TP: number of change pixels correctly detected; FP: number of no-change pixels incorrectly labelled as change; TN: number of no-change pixels correctly detected; FN: number of change pixels incorrectly labelled as no-change.

Table 2 shows the results in terms of the correct classification for the seven experiments. The final result for each experiment is averaged by the number of pairs of images processed. The number of iterations used in our FCM (k_{max}) is set to the number of iterations where HNN gained the convergence for each set of experiments, i.e. E1, E3 = 4, E2, E5 = 8, E4 = 10 and E6, E7 = 5.

Table 2 Averaged PCC scores for each method against the set of experiments

$x10^{-3}$	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**	**E7**
BRU	921	821	945	653	698	819	615
HNN	987	943	991	789	876	901	847
FCM	944	954	956	823	901	848	844

From results in table 2, one can see that *FCM* improves the performance of *HNN* for experiments E2, E4 and E5 where the number of iterations is higher than the used for the other experiments. This means that the *FCM* approach is suitable for images where the number of iterations is high. The *FCM* outperforms the other approaches when each image in the pair displays high variability due to different illumination conditions as in the experiments E2, E4 and E5. This means that the *FCM* should be applied in image sequences captured under such illumination conditions where it is foreseeable that the number of iterations could become high.

The best performance achieved by the *FCM* approach against *HNN* in those experiments, can be interpreted in the light of the mutual influence between two nodes based on the relevance's values. Indeed, as the number of iterations increases, the relevance of each node achieves higher stability (less number of changes in the activation level). *FCM* and *HNN* achieve both a similar performance for E7 (with noise).

6.3 Performance of FCMs for Stereovision Matching

In stereovision matching, the proposed Fuzzy Cognitive Maps strategy (FCMS) is compared against the Support Simulated Annealing (SANN) in Pajares and Cruz (2004), but also with the method described in Pajares et al. (2000) which is a Relaxation Labeling (RELB) approach and the method described in Pajares et al. (1998), which is an optimization approach based also on the Hopfiel Neural Network (HNNB1).

SANN applies the similarity constraint through Support Vector Machines and RELB and HNNB1 both apply the similarity constraint by computing a matching

probability based on the estimation of a probability density function through the Bayes's theory. The matching probabilities are used as the inputs for the relaxation and optimization processes respectively. From these processes, RELB performs an iteration procedure by applying smoothness, ordering and uniqueness constraints. HNNB1 performs the optimization process by mapping the smoothness and uniqueness constraints in an energy function which is to be minimized. From HNNB1, we have implemented a new version HNNB2, by mapping the ordering constraint as an energy function to be minimized and applying the similarity constraint as the 4-dimensional difference null vector $\boldsymbol{x}$. HNNB2 can be considered a very close approach to the described in Ruichek and Postaire (1996), although this work uses edge pixels as features, we have modified the original method in Ruichek and Postaire (1996) to use edge-segments as in SANN.

We have also compared our approach with the Stochastic Stereovision Matching Method (SSVM) in Barnard (1989), also used in Hattori et al. (1998). This method uses the regularization criterion proposed in Poggio et al. (1985), where an energy functional is minimized based on a penalty functional which measures the dissimilarity between corresponding features (similarity constraint) and a stabilizing functional by which the smoothness constraint is imposed. The energy minimization is carried out through the Simulated Annealing algorithm, we have used a value of 50 as in Barnard (1989) for the regularization parameter λ (this works well for images quantized in 8-bit values) and the same neighborhood criteria as that used in this paper. Two differences are considered in this implementation with respect to our implementation: 1) the edge-segments disparities are the outputs obtained in SSVM, which are used to obtain the correspondences and 2) the hierarchical coarse-to-fine control structure with re-heating in Hattori et al. (1998) is not used in our implementation.

Finally we have chosen the Minimum Differential Disparity algorithm (MDDA) in Medioni and Nevatia (1985) for comparative purposes for the following reasons: 1) it is a merit relaxation approach, 2) it applies the commonly used constraints (similarity, smoothness and uniqueness); 3) it uses edge segments as features and the contrast and orientation of the features as attributes; and 4) some concepts of MDDA, such as minimum differential disparity, overlapping concept, disparity limit or average disparity are used in our FCMS approach. Table 3 summarizes the main differences between the eight strategies compared. All methods use edge-segments as features and the same four attributes.

We have selected 82 stereo pairs of realistic stereo images from an indoor environment. All tested images are 512 x 512 pixels in size, with 256 gray levels. Of all the possible combinations of pairs of matches formed by pairs of segments from left and right images only 1701 are considered for testing.

Table 4 displays the percentage of successes for each method (FCMS, SANN, RELB, HNNB1, HNNB2, SSVM, and MDDA) as a function of the number of iterations. These values are averaged over the above full number of stereo pairs considered for matching. Iteration 0 corresponds to the results obtained only by applying the similarity constraint, i.e. before the iteration process is triggered. This is the starting point for the FCM process proposed in this approach. Intermediate results are also displayed for a number of iterations of 15. The final results correspond to a

number of iterations of 30, which is on average the number of iterations required for the FCMS for convergence with the 82 pairs of stereo images used for testing.

Table 3 Summary of Stereovision Matching Methods and Constraints

	Stereovision matching constraints				
	Similarity	**Smoothness**	**Ordering**	**Epipolar**	**Uniqueness**
FCMS	Simple difference vector	mapped as coefficients aggregated in the causal weight between concepts	mapped as coefficients aggregated in the causal weight between concepts	mapped as coefficients aggregated in the causal weight between concepts	applied by selecting the highest causal concept values
SANN	Support Vector Machines	mapped as an energy minimized by Simulated Annealing	mapped as an energy minimized by Simulated Annealing	mapped as an energy minimized by Simulated Annealing	applied by selecting the highest state values
RELB	Bayes probability density estimation	Probabilistic relaxation	Probabilistic relaxation	mapped under the overlapping concept	applied by selecting the highest probabilities
HNNB1	Bayes probability density estimation	mapped as an energy minimized by Hopfield	No	mapped under the overlapping concept	mapped as an energy minimized by Hopfield
HNNB2	Euclidean distance without estimation	mapped as an energy minimized by Hopfield	mapped as an energy minimized by Hopfield	mapped under the overlapping concept	mapped as an energy minimized by Hopfield
SSVM	mapped as an energy minimized by regularization	mapped as an energy minimized by regularization	No	implicit application by image registration	No
MDDA	qualitative boolean function	merit function relaxation	No	implicit application by image registration	applied by selecting the highest merits

Table 4 Percentage of successes against the number of iterations

Iteration #	**0**	**15**	**35**
FCMS	69.3	81.1	94.2
SANN	76.1	83.3	92.1
RELB	74.5	82.1	90.2
HNNB1	70.4	81.6	90.2
HNNB2	67.8	69.3	67.8
SSVM	0	39.2	57.4
MDDA	66.1	68.7	69.7

According to values from table 2, the following conclusions may be inferred:

1) *iteration 0*: the best performances are achieved with the methods that apply a learning process, i.e. SANN, RELB and HNNB1. The methods without previous estimation or learning (FCMS, HNNB2 and MDDA) obtain the worst results at this phase.

2) *Iteration process*: As the number of iterations progresses all methods achieve better performances; this means that the mapping of different constraints is positive. The ordering constraint is not decisive: HNNB2 (with ordering) obtains worse results than HNNB1 and HNNP (without ordering). In Pajares and Cruz (2004) is reported that SANN reaches its equilibrium with an average of 65 iterations, with such number of iterations the performance of SANN is comparable to the performance of FCMS with 35 iterations. Therefore, we can conclude that the fuzzy causal approach under FCMs is suitable in terms of performance in stereovision matching, i.e. the causal reasoning and decision process under a fuzzy point of view is consistent with that of the human. This is the reason for the better performance of our FCMS approach against the remainder methods, particularly against the SANN approach. Another important reason comes from equation (1), where the previous causal concept $A_i(t)$ contributes to the updating of the current one $A_i(t+1)$, this implies that the network in the FCMS achieves a rapid stabilization. Finally, SANN only reaches the 96.0 in percentage with 400 iterations, as reported in Barnard (1989). MDDA obtains the worst results.

7 Conclusions

We have applied the Fuzzy Cognitive Maps framework for three computer vision tasks, verifying its performance as compared to other existing methods. Starting from the general definition we can adapt and customize the different parameters and coefficients involved on it for each specific approach. This means that FCMs become an excellent tool when spatial relations among elements in the images can be established based on the Gestalt's principles. In the proposed approaches the elements used were pixels in image classification and change detection and edge segments in the stereovision matching approach. This establishes the base for its application to other different image computer vision problems, where the only requirement is the mapping of the spatial relations or properties.

The main drawback of FCMs is its computational cost due to the iterative process applied. But this problem also appears in the remainder iterative approaches. In the future, this could be improved through parallel implementations.

Another research line will come from the study of the non-linear functions for computing the activation levels in equations (2) to (4), so that the convergence process can be accelerated without loss of performance.

References

Barnard, S.T.: Stochastic stereo matching over scale. Int. J. Comput. Vision 3(1), 17–32 (1989)

Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer, Plenum Press, Dordrecht (1981)

Bruzzone, L., Fernández-Prieto, D.: Automatic Analysis of the difference Image for unsupervised change detection. IEEE Trans. Geoscience Remote Sensing 38(3), 1171–1182 (2000)

Cabrera, J.B.D.: On the impact of fusion strategies on classification errors for large ensambles of classifiers. Pattern Recognition 39, 1963–1978 (2006)

Duda, R.O., Hart, P.E., Stork, D.S.: Pattern Classification. Wiley, Chichester (2000)

Haykin, S.: Neural Networks: A comprehensive Foundation. Mcmillan College Publishing Co., New York (1994)

Hattori, S., Okamoto, A., Hasegawa, H.: Stereo matching by simulated annealing incorporating a diffusion equation. In: Proc. ASPRS 1998 Annual Conference, pp. 1030–1041 (1998)

Huertas, A., Medioni, G.: Detection of Intensity Changes with Subpixel Accuracy Using Laplacian-Gaussian Masks. IEEE Trans. Patt. Anal. Machine Intell. 8(5), 651–664 (1986)

Kapur, J., Sahoo, P., Wong, A.: A new method for gray-level picture thresholding using the entropy of the histogram. Computer Vision Graphics Image Processing 29(3), 273–285 (1985)

Kittler, J., Hatef, M., Duin, R.P.W., Matas, J.: On Combining Classifiers. IEEE Trans. Patt. Anal. Machine Intell. 20(3), 226–239 (1998)

Koffka, K.: Principles of Gestalt Psychology. Harcourt, New York (1935)

Kosko, B.: Fuzzy Cognitive Maps. Int. J. Man Machine Studies 24, 65–75 (1986)

Kosko, B.: Neural Networks and Fuzzy Systems: a dynamical systems approach to machine intelligence. Prentice-Hall, NJ (1992)

Krotkov, E.P.: Active Computer Vision by Cooperative Focus and Stereo. Springer, New York (1989)

Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley, London (2004)

Leu, J.G., Yau, H.L.: Detecting the dislocations in metal crystals from microscopic images. Pattern Recognition 24, 41–56 (1991)

Lew, M.S., Huang, T.S., Wong, K.: Learning and Feature Selection in Stereo Matching. IEEE Trans. Patt. Anal. Machine Intell. 16(9), 869–881 (1994)

Medioni, G., Nevatia, R.: Segment Based Stereo Matching. Computer Vision, Graphics and Image Processing 31, 2–18 (1985)

Miao, Y., Liu, Z.Q.: On Causal Inference in Fuzzy Cognitive Maps. IEEE Trans. Fuzzy Systems 8(1), 107–119 (2000)

Nevatia, R., Babu, K.R.: Linear Feature Extraction and Description. Computer Vision, Graphics, and Image Processing 13, 257–269 (1980)

Pajares, G., Guijarro, M., Herrera, P.J., Ribeiro, A.: Combining Classifiers through Fuzzy Cognitive Maps in natural images. IET Computer Vision 3(3), 112–123 (2009)

Pajares, G., Sánchez-Beato, A., de la Cruz, J.M., Ruz, J.J.: A neural Network Model for image Change Detection Based on Fuzzy Cognitive Maps. In: Martí, J., Benedí, J.M., Mendonça, A.M., Serrat, J. (eds.) IbPRIA 2007. Part I, LNCS, vol. 4477, pp. 595–602. Springer, Heidelberg (2007)

Pajares, G., Cruz, J.M.: Fuzzy Cognitive Maps for stereovision matching. Pattern Recognition 39, 2101–2114 (2006)

Pajares, G.: A Hopfield Neural Network for Image Change Detection. IEEE Trans. Neural Networks 17(5), 1250–1264 (2006)

Pajares, G., Cruz, J.M.: On Combining Support Vector Machines and Simulated Annealing in Stereovision Matching. IEEE Trans. System, Man Cybernetics, Part B 34(4), 1646–1657 (2004)

Pajares, G., Cruz, J.M., López-Orozco, J.A.: Relaxation labeling in stereo image matching. Pattern Recognition 33, 53–68 (2000)

Pajares, G., Cruz, J.M., Aranda, J.: Relaxation by Hopfield network in stereo image matching. Pattern Recognition 31(5), 561–574 (1998)

Poggio, T., Torre, V., Koch, C.: Computational Vision and regularization theory. Nature 317, 314–319 (1985)

Rosin, P.L., Ioannidis, E.: Evaluation of global image thresholding for change detection. Pattern Recognition Letters 24, 2345–2356 (2003)

Ruichek, Y., Postaire, J.G.: A neural network algorithm for 3-D reconstruction from stereo pairs of linear images. Pattern Recognition Letters 17, 387–398 (1996)

Scharstein, D., Szeliski, R.: A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms. Int. J. Computer Vision 47(1/2/3), 7–42 (2002)

Shorliffe, E.H.: Computer-Based Medical Consultations: MYCIN. Elsevier, NY (1976)

Tanaka, S., Kak, A.C.: A Rule-Based approach to binocular stereopsis. In: Jain, R.C., Jain, A.K. (eds.) Analysis and Interpretation of range images, pp. 33–139. Springer, Berlin (1990)

Tsardias, A.K., Margaritis, K.G.: Cognitive Mapping and Certainty Neuron Fuzzy Cognitive Maps. Information Sciences 101, 109–130 (1997)

Tsardias, A.K., Margaritis, K.G.: An experimental study of the dynamics of the certainty neuron fuzzy cognitive maps. Neurocomputing 24, 95–116 (1999)

Tsardias, A.K., Margaritis, K.G.: The MYCIN certainty factor handling as uniform operator and its use as threshold function in artificial neurons. Fuzzy Sets and Systems 93, 263–274 (1998)

Wang, D.: The time dimension for scene analysis. IEEE Trans. Neural Networks 16(6), 1401–1426 (2005)

Wu, Q.Z., Cheng, H.Y., Jeng, B.S.: Motion detection via change-point detection for cumulative histograms of ratio images. Pattern Recognition Letters 26, 555–563 (2005)

Xirogiannis, G., Chytas, P., Glykas, M., Valiris, G.: Intelligent impact assessment of HRM to the shareholder value. Expert Systems with Applications 35, 2017–2031 (2008)

Xirogiannis, G., Glykas, M.: Intelligent modelling of e-business maturity. Expert Systems with Applications 32, 687–702 (2007)

Xirogiannis, G., Glykas, M.: A soft knowledge modeling approach for geographically dispersed financial organizations. Soft Computing 9, 579–593 (2005)

Xirogiannis, G., Glykas, M.: Fuzzy Cognitive Maps in Business Analysis and Performance-Driven Change. IEEE Trans. Engineering Management 51(3), 334–462 (2004)

Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decision making. IEEE Trans. System Man and Cybernetics 18(1), 183–190 (1988)

Yaman, D., Polat, S.: A fuzzy cognitive map approach for effect-based operations: an illustrative case. Information Sciences 179, 382–403 (2009)

Zimmermann, H.J.: Fuzzy Set Theory and its applications. Kluwer Academic Publishers, Dordrecht (1991)