Acta Universitaria
DIRECCIÓN DE APOYO A LA INVESTIGACIÓN Y AL POSGRADO

Object Tracking based on Fuzzy Color Blobs.

Francisco-Javier Montecillo-Puente * and Victor Ayala-Ramirez**

ABSTRACT

One of the mayor goals in computer vision is object representation. Object representation aims to determine a set of features that best represents a specific object in an image, for example interest points, edges, color and texture. However, objects are generally composed of several regions containing different information which is more or less convenient to be represented by one of these features. Furthermore, each of these regions could be static or moving with respect to each other. In this sense, this paper presents an object representation based on fuzzy color blobs and spatial relationships among them. This approach of object representation is used to track rigid and articulated objects.

RESUMEN

En visión por computadora una de las metas principales es la representación de objetos. Dada una imagen, la representación de objetos intenta definir un conjunto de características que mejor los representa a partir de la información visual. Por ejemplo, se han utilizado puntos de interés, bordes, color y textura. Sin embargo, los objetos están formados por diferentes regiones con información variada que es más o menos conveniente a ser representada por alguna de las características antes mencionadas. Además, cada una de estas regiones puede estar estática o en movimiento respecto al resto de las regiones que componen el objeto. En este sentido, este trabajo presenta una representación de objetos basadas en regiones de color difuso y sus relaciones espaciales. Este enfoque es utilizado para el seguimiento de objetos rígidos y articulados.

Recibido: 10 de Enero del 2012
Aceptado: 14 de Febrero 2012

INTRODUCTION

In the literature related to object representation,objects has been represented by interest points [1], color blobs [2] [3], texture [4], templates [5], active contours [6], geometric shapes [7], among others. Due to the wide range of objects can be present in a scene (trees, books, phones, people, among others), these representations are adapted generally to modeling a specific kind of objects or regions. We are interested in color blobs, in particular in homogeneous colored regions. In [3] a variant of the mean shift algorithm is used to track faces, this method continuously tries to adapt the center clusters based on the current images, one of the main inconveniences is the number of parameters required to tune for different scenarios. The HSV color space, (Hue, Saturation, Value) was used to represent the color features. In [2], the authors propose to determine color blobs based on a clustering technique similar to the k-means [8] and displacement vectors. Objects are represented by a combination of color blobs and motion vectors. In [9] has been proposed that a virtual scene can be represented by the spatial relations among objects.

Palabras clave:
Regiones de color; conjuntos difusos; seguimiento de objetos; objeto rígido.
Keywords:
Color blobs; fuzzy sets; object tracking; rigid object.

Furthermore, relationships has been used to model the human body based on pictorial cues [10]. In this work, each pictorial cue was related to a body part. Additionally, spatial relationships were used among cues to be coherent with the human body.This paper is on object representation based on color blobs and the spatial relationships among them. First, the number of blobs in the object is determined by using the mean shift algorithm [11], then each of the identified colors is represented by fuzzy rules [12]. In the next step, the colored regions are analyzed to extract its spatial relationships by using the histogram of forces [13].

*Business Management and Digital Arts Department, DICIS, Universidad de Guanajuato, carr. Salamanca-Valle km. 3.5+1.8, com. Palo Blanco, 36700 Salamanca, Mexico E-mail fmontecillo@ugto.mx
**Electronics Engineering Department, DICIS, Universidad de Guanajuato, carr. Salamanca-Valle km. 3.5+1.8, com. Palo Blanco, 36700 Salamanca, Mexico E-mail ayalav@ugto.mx

This histogram gives a measure of the angle between two areal objects. The proposed object representation allow us to track objects and to determine if these are rigid or articulated. The content of the paper is organized as follows. Firstly, we present an overview of the main stages of the proposed object representation is given. The mean shift algorithm is reviewed in second place, and some tests are presented too. Then, the fuzzy color representation for color blobs is presented, and it is continued by the development of the force histogram and the spatial relationships. Afterwards, the results and applications are presented and we end by presenting our conclusions.

OUR PROPOSAL

The kind of objects used in this paper are composed of several colored regions, which are commonly found in several indoor and outdoor scenes. Our proposal exploits these object characteristics to represent object as color blobs and their spatial relationships. The proposed representation includes the following stages which are developed further,

1. Number of colors: first a region is selected from a image which contains a object to be represented, the number of colors which composed the object is determined by using the mean shift algorithm [11].
2. Fuzzy color: from each region determined by the mean shift algorithm, membership functions are computed to represent color by a fuzzy rule.
3. Spatial relationships: the histogram of forces and fuzzy rules are used to determine the relationships among color blobs.
4. The final representation of the object is defined by color blobs and its spatial relationships.

The proposed method is evaluated in the following scenarios: 1) a tracking system was implemented to test the fuzzy color representation; 2) a rigid object was tracked; and 3) the property on non rigid object was tested over a sequence of images including an articulated object.

MEAN SHIFT

An still research problem is clustering, which consists in grouping a set of n elements of any dimension. The methods generally used to solve this problem are k-means, fuzzy C means, ISODATA, merging, splitting and mean shift, a brief discussion of those is found in [14]. Almost all clustering methods require as input the number of clusters, one exception to this is the mean shift algorithm. However, the mean shift algorithm requires three parameters which are sensitive to the number of clusters the algorithm determines. This algorithm [11] consists in determining the number of modes in a density function. First, a collection of records to analyze is defined. For example a set $\{\mathbf{X}_i\}$ with n data in a d-dimensional Euclidean space ($\mathbb{R}^d$). Over these data, an estimated multivariate density function $\hat{f}$ using a kernel function $K(\mathbf{x})$ with window radius h at the point $\mathbf{x}$ is defined as

$$\hat{f}(\mathbf{x}) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{\mathbf{x}-\mathbf{X}_i}{h}\right) \tag{1}$$

Then, this function and its gradient are used by the mean shift algorithm to determine its number of nodes. Generally, the Epanechnikov's kernel is used to obtain a Mean Integrated Squared Error on the estimation. Second, the method selects randomly a set of points. These point locations are the first candidates to modes. At each candidate mode location, the gradient of the density function is evaluated and used to displace it towards a next close location with maximum density. If a set of these candidates are close enough, these are merged into one, i.e. using the median of its location. The criteria to stop the algorithm is determined if the number of iterations is reached or when the candidate locations remain the same in two consecutive iterations. In practical implementation three parameters are required, T_1 which is the minimum number of points required in a cluster candidate, h which is the radius of a hyper-sphere (the Epanechnikov's kernel), and a parameter codifies the fact that a valley must exist between each pair of cluster candidates, T_2. Considering the locations connecting two candidate clusters as candidates to clusters, T_2 is the number maximum each must contain. If a valley doesn't exist the candidate with minimum density must disappear. Among the parameters, it have been reported that the radius of the hyper-sphere h is the most sensitive in final number of modes. Each final mode is considered a center of a cluster in the analyzed data.

Parameters Evaluation

We performed some tests to determine practical values for the mean shift parameters. In our, implementation the parameter T_2 was not considered, because the computation time increased considerably. Moreover, we observed that for the kind of objects used T_2 has not much influence on the results. Hence, the performed test only evaluates the influence of T_1, h and the number of iterations in respect of the number of clusters detected.

Acta Universitaria
DIRECCIÓN DE APOYO A LA INVESTIGACIÓN Y AL POSGRADO

A Color blob consists of a set of pixels whose color coordinates are close enough in some sense. This situation can be emulated by defining homogeneous colored images and adding them Gaussian noise of variance σ. The images shown in Fig. 1 were used to determine practical values on T_1, h and number of iterations. The image in Fig. 1 (a), consists of four homogeneous strips with different RGB colors, that is their distance between each other color is enough to be easily separated. The image in Fig. 1 (b) consist of four strips with colors close among them, in particular close to (0,127,0) RGB coordinates.

The image with different colors was used to determine the conditions when new color appears as the variance increases, while the image with close colors was used to detect when merging among colors starts. The test images were created artificially in RGB color space, but the CIELab color space was used to perform the tests. The color in CIELab space is represented by three components: a lightness component(L) and two color components (a and b). These images are of 360×360 pixels and the Gaussian noise was added in RGB space before transforming to CIELab.

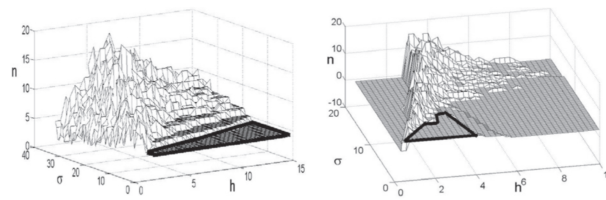
(a) (b)

Figure 1 . Test images to determine the convenient values for T_1 and h parameters.

The first test consists in evaluating the h parameter in respect of the number of colors. In this case, we fixed the number of iterations to 200 and T_1 to 50. The plots shown in Fig. 2 resumes the results, where σ is the noise variance, h is the radius of the hypersphere and n the number of colors detected. On there plot, a polygonal region is marked, this region represents the values of h in the algorithm which finishes with the correct number of colors.

In the case of the image with different colors, refer to Fig. 1 (a), it is noted that the number of colors increases as the variance increases as σ increases, Fig. 2 (a). That is, each color strip is split into several colors. Also, we can observe that high values of σ limits the range of values h on which the correct number are properly detected.

The plot in Fig. 2 corresponds to the results for the case of close colors. In this plot, we can observe a limited range of values from 1 to 4.0 for the parameter h within the correct number of colors is properly detected. In fact, increasing the value of h, the colors start merging. Moreover, we note that large values of σ has the effect of increasing the number of colors. This situation is overcame by increasing the value of h. Nevertheless, care must be taken because of the merging effect.

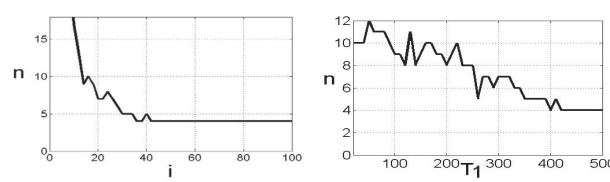


(a) (b)

Figure 2 . Plots of noise variance σ, number of colors detected n and h parameter for the images.

The second test consists in evaluating the number of iterations and the parameter T_1. The results for the case of different colors are shown in Fig. 3. We added Gaussian noise with $\sigma = 6.5$, and fixed $h = 2.5$ and $T_1 = 100$ in the mean shift algorithm. The plot in Fig. 3 (a) shows the number of iterations i versus the number of color detected n. In this plot, we observe that the number of iteration required to determine the four clusters was about 43. Choosing values for h and T_1 on which the number of colors are properly detected, the number of iterations is close to 40.

In the Fig. 3 (b) a plot of the parameter T_1 versus the number of colors detected is shown. It is noted that algorithm is very sensitive in respect to T_1. That is, with small values of T_1 it found more colors. However, as its value increases, the number of clusters remains the same. In this case for a T_1 value greater than 430. Performing similar tests on different images sizes, it was found that giving a value between 0.5% and 1% of total number of pixels to T_1 almost always the correct number of clusters was determined.



(a) (b)

Figure 3 . These plots present the results of the mean shift algorithm for the image with different colors:(a) number of colors versus iterations, (b)number of colors versus parameter T_1.

FUZZY COLOR

In fuzzy logic and fuzzy sets theory, a fuzzy set is a set in which each element has some degree of membership. If the element is not included in of the set this element has zero membership, if the element is included, the element can take a value in the interval $(0, 1]$. The shape of a function called membership functions define the assignation of membership to each element in the set. In order to combine several fuzzy sets, there exists operators and a IF-THEN rule. The operators include AND, OR and NOT [12]. The α–cut operator modifies the degree of membership assignment by setting to zero all membership values which are below the α value, while others remain the same. These concepts are to be used to define the color representations of blobs.

One of the well known problems in color based approaches to represent objects is light variation. Generally, these variations degrade the performance of the representation. To overcome this problem, we used the CIELab color space to obtain light and chroma information, then applied fuzzy rules to define color blobs, and finally we applied a simple updating method.

The procedure to determine the fuzzy color representation of blobs is based on the the center clusters found by the mean shift algorithm, see Section 3 . This fuzzy color representation is achieved as follows,

1. For each each cluster and color component, compute its histogram.
2. Assign a membership function that fits the histogram. Usually, triangular, trapezoidal, normal membership functions are used.
3. For a specific color, each membership function defines a fuzzy set.
4. Then fuzzy rules are evaluated to determine if a colored pixel is part of a target color, that is

if $(R_L$ and R_a and $R_b)$ **then** p is part of color target

where R_L, R_a y R_b are defined as

R_L: L_p is membership grade of the target fuzzy set L
R_a: a_p is membership grade of the target fuzzy set a
R_b: b_p is membership grade of the target fuzzy set b

The updating method is used to displace the location of the membership functions, as the light conditions varies. In particular, a triangular membership function (Fig.4) is updated by displacing its center location p_1 while the distances from p_1 to p_0 and p_2 are kept fixed. That is,

$$p_1' = p_1 + \gamma \Delta p \tag{2}$$

where p_1 is the current center location, p_1' is the next center location, γ is a smoothing constant, and Δp is an updating value.

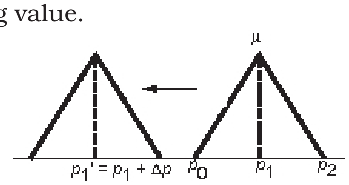


Figure 4 . Effect of updating a triangular membership function. Triangular membership functions are defined by three parameters p_0, p_1 and p_2.

As an example, for the a component, Δp is computed as

$$\Delta p = \frac{1}{n} \sum_{i=1}^{n} P_i^a - p_1 \tag{3}$$

where P_i^a is the a component of i-th pixel with membership value higher than a predefined α-cut, n is the number of pixels satisfying the α-cut.

In the Fig. 5 it is shown an example of light varying conditions. These plots shown the variation of the triangular membership function as light condition varies. In this example, lights were turned-off and turned-on several times in the indoor workspace. These effects could be seen in the oscillatory parts of the plot. In this example, γ and α-cut were set to 0.2 and 0.75, respectively.

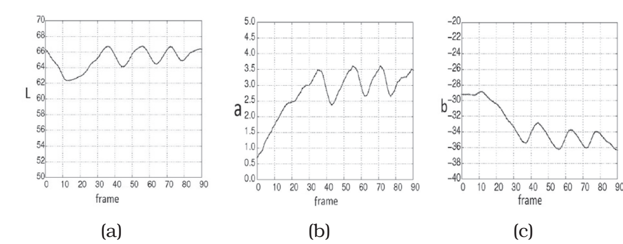


Figure 5 . Variation of light conditions and the adaptation of membership functions for a blue colored object.

SPATIAL RELATIONS

Spatial relationships among objects aims to quantify or qualify theirs relative spatial information. Particularly,orientation relationships has been proved to be useful for navigation and symbol matching [15][16].

The orientation relation between two objects A and B tries to measure the following proposition,

object A is in direction θ of object B

Several approaches has been used to evaluate it. The authors in [17] assume a pair of objects could be represented as points. Hence, their relative orientation is given by computing the orientation of the line joining the centroid of the objects.However, this simple method has some disadvantages, the most important is that the shape of the objects is completely omitted.

In order to take into account shape, histogram of angles were proposed [18]. This is calculated by computing all the angles formed for every pair of points, one in the object B and one in the object A. The representative angle between objects is given by the angle with highest frequency or could be computed by analyzing this histogram. However, its disadvantage is its high computational cost. So for on-line applications it is not convenient.

Histogram of Forces

Force histograms is more efficient and general method to compute the relative orientation between objects [13]. This method takes into account the shape, size and distance between objects. Initial objects are defined as a non empty set of pixels E. Given an angle θ, the object can be represented by a finite number of oriented line segments, each in θ direction. An oriented line is defined as $\Delta_\theta(v)$ where θ is the orientation of the line and v is a vector parallel to this line. So the line segmented object is $E \cap \Delta_\theta(v)$, denoted as $E_\theta(v)$. Assuming there are two objects A and B, their line segmented versions respect to an angle θ are expressed as $A_\theta(v)$ and $B_\theta(v)$, respectively. The evaluation of the proposition is achieved by evaluating a function $F(\theta, A_\theta(v), B_\theta(v))$. This function takes as arguments an angle θ and a pair of line segmented objects. Line segments that contribute on the affirmation of this proposition are the ones in both objects that are over the same oriented line. The evaluation of F at the given orientation θ is the total contribution for each pair of segments.

Given a pair of objects A and B,a function F and taking into account all possible values for θ, 0-360^o,: a function F^{AB} is called the F-histogram associated to the objects A and B. This function has a value for every angle. The maximum value in this function corresponds to the angle θ that more satisfies the proposition . In this work, the spatial relationships are evaluated by using this method [13].

Rigidity Detection

In order to determine if in a sequence of images a set blobs are rigidly attached each other, each color blob is tracked and the force histograms among them are computed. Then, relative orientations among objects are obtained from these histograms. After recording relative orientation over a number of frames, the orientation trajectories are analyzed as follows,

1. Set the trajectories to a common reference to be comparable, i.e. shift each trajectory to a zero initial orientation.
2. Obtain the average orientation trajectory $Av(i)$ over all the trajectories.
3. Compute an average error curve, at frame i the error $E(i)$ is determined by

$$E(i) = \frac{1}{n}\sum_{k}^{n}(Av(i) - O^k(i)) \tag{4}$$

   where n is the number of orientation trajectories, O^k represents the k-th orientation trajectory between a pair of objects.

4. Compute the absolute mean error over the average error curve, if the mean error is below a predefined threshold, the objects are considered rigidly attached.

EXPERIMENTAL RESULTS

In this section, the evaluation of the object representation is presented. In order to show the feasibility of the system, a tracking system was implemented which includes: fuzzy color blobs, spatial relationships, fuzzy color update and a Kalman Filter. Kalman filtering was used to predict the location of each blob and the relative orientations between objects. Three scenarios were analyzed to test the proposed object representation. First, a sequence of amateur soccer players was used to test the fuzzy color model. In the second scenario a rigid object is analyzed, and finally in the third scenario an articulated object was studied.

Fuzzy Color Blob Tracking

To build this scenario, the sequence shown in Fig. 6 was used. In this scenario, the goalkeeper in the image sequence was tracked. The test, includes fuzzy color tracking model, the simple updating step and the kalman filtering to predict the image position of the goalkeeper. In this sequence, the object was well tracked in spite of some dust because of the players, the clouds and the goalkeeper motion.

Figure 6 . Goal keeper sequence to test the fuzzy color tracking, a black line is marked to define the boundary of the object.

Rigidity between Color Blobs

The first test consists in determining if three Lego blocks are rigid or not. In this case a logo setup was realized. This setup is a motor turning at constant speed with an object attached to it. The object was composed of three Lego blocks linked each other using rigid beams, Fig. 7.

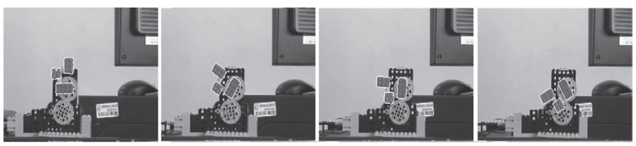


Figure 7 . A sequence of Lego blocks rigidly linked, a white line is marked to define the boundary of each block.

To illustrate the procedure of rigidity detection, two relationships were used. Each relationship is on a pair of different blobs composing object. The plot shown in Fig. 8 presents the results of computing the spatial relationships and the error between them.It was found, the absolute mean error is 0.4284 degrees, this low value indicates the rigidity connection among parts of the object.

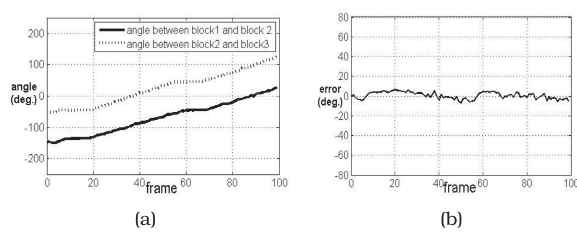


Figure 8 . Spatial relationships plots for a rigid object: (a) orientation between objects, (b) relative error.

No-Rigidity Between Color Blobs

In a similar way as rigidity detection, non rigidity is detected if the mean error between angle curves is above a threshold. The sequence used in this scenario is shown in Fig. 9, also two spatial relationships were defined.

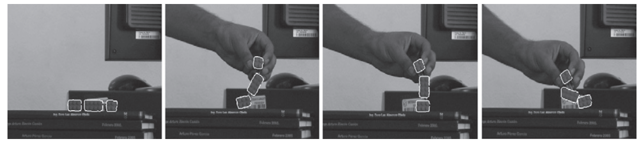


Figure 9 . A sequence of Lego blocks articulated, a white line is marked to define the boundary of each block.

The plots of curve angles over a number of frames and its error are shown in Fig. 10. The resulting absolute mean error is of 34.2 degrees, which is considerably bigger than its rigidity counterpart.

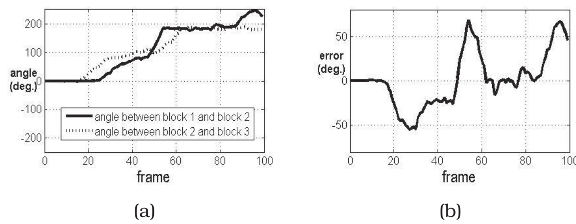


Figure 10 . Spatial relationships plots for a non rigid object: (a) orientation between objects, (a) relative error.

Fuzzy Color Blobs and Force Histogram Evaluation

The evaluation of the pertinence of the fuzzy color representation and spatial relationships was achieved as follows,

1. A setup was built, which consisted of a pair of colored circles rigidly linked and attached to a motor.
2. The setup was calibrated to rotate at each step an angle of 4.5^o.
3. Thirty sequences were captured, each image consisted of a successive 4.5^o angle rotation until a full turn was completed.
4. For each sequence the regions were tracked and its angle curve was computed using maximum of the histogram of forces at each frame.
5. An average curve was obtained form all the angle curves.
6. Then the absolute mean error was computed, its value is of 1.7713.

Computation Time Evaluation and Implement

In the table 1, it is shown the time required at each stage for the rigidity related sequences. From this table, it is noted the most consuming time stage is the force histogram computation. Total time computation was about 20 ms. This was reduced to 10 ms. However, some workarounds were implemented in the histogram of forces stage. First, to compute the line segmented version of the objects a look-up tables of oriented lines was computed. That for each orientation an image of the size of the objects is built, whose content are black and white lines spaced by some predefined number of pixels. However, if the space between two lines is bigger the accuracy of the computed angle is affect. So a trade-off between the computing time and the accuracy must be taken into account dependent of the application. Another workaround was the use of a Kalman Filtering to predict the orientation of the objects. The gain with the filtering is to compute partial histograms located around the predicted orientation. These works around are enough to reduce the computation time to only $6ms$. The computation were performed on a Laptop DELL, 4GB RAM, Core i7 1.6GHz.

Table 1 .

Time required per stage.

stage	time (ms)
Blob color location	5.0
Force histograms	16.0
Kalman filtering and others	0.5

CONCLUSION

In this paper it was presented a to detect color blobs based on the mean shift algorithm and fuzzy sets. The mean shift algorithm was used to detect the number of color blobs. However, for the mean shift algorithms it is important to tune two parameters in respect to the application. Instead, a study about the variation of these parameters was presented. Within this study the feasible regions for these sensitive parameters were experimentally computed. With the use of the histograms of forces it was possible to determine the spatial relationships between color blobs. This feature that almost is always omitted in object representation allow us to determine if objects were rigidly connected or not. That is, objects are represented by color blobs and its spatial relationships. This combination of fuzzy color blobs and spatial relations ships allow us to implement an on-line tracking system. It was seen that spatial relationships are the most expensive computations, however it was possible to reduce its computation in 50 percent at expense of the accuracy in the computations. The final time required per cycle was close to 10ms for tracking three color objects including two computational relationships. An initialization stage is required for the tracking system that takes 10ms, and the user must interact with the system to select the region were the object is located. Finally, even if color blob was used another kind of features like texture, depth, among others, could be used. In fact, a mixed of blob color, texture, depth and spatial relationships could be implemented to represent objects. One of the main disadvantages of blob color based tracking methods is the number of parameters to tune. In general, these include: the number of color blobs , a threshold criteria to determine if a pixel is part or not of a color blob, and updating constants. In this paper, to define the number of colors the mean shift was used, however this method depends on three parameters. These parameters where studied to define the range of values over which the method works properly. The fuzzy rule we used to define the color blobs requires a parameter, the α−cut; additionally our method requires an adaptation factor. These two parameters are the main drawbacks of our method. The values of these parameters depend on several situations, mainly, the quality of the camera sensor, light conditions (very sensitive in darkness), and surface of the objects (i.e. textured objects). These parameters have to be tuned manually depending on these conditions.

ACKNOWLEDGMENT

This work was partially supported by Mexican program NPTC-PROMEP-SEP.

REFERENCES

[1] Lowe, D. G. (1999). Object recognition from local scale-invariant features. *In in Proceedings of IEEE International Conference on Computer Vision,* 1150--1157.

[2] Heisele, B., Ritter, W. (1995). Obstacle detection based on color blob flow. *In Intelligent Vehicles '95 Symposium., Proceedings of the,* 282 --286.

[3] *Open Source Computer Vision Library Reference Manual.*

[4] Lazebnik, S., Schmid, C., Ponce, J. (2006). A discriminative framework for texture and object recognition using local image features. *Lecture Notes in Computer Science* 4170:423--442.

[5] Brunelli, R. (2009). Template Matching Techniques in Computer Vision: Theory and Practice. *Wiley.*

[6] Blak, A., Isard, M. (2000). Active Contours. *Springer.*

[7] Petitjean, S. (1995). Algebraic geometry and object representation in computer vision. *In M. Hebert, J. Ponce, T. Boult, A. Gross, editors, Object Representation in Computer Vision, volume 994 of Lecture Notes in Computer Science,* 155--165. Springer Berlin / Heidelberg.

[8] Hartigan, J. A., Wong, M. A. (1979). A k-means clustering algorithm|. *Applied Statistics* 28:100--108.

[9] Fisher, M., Savva, M., Hanrahan, P. (2011). Characterizing structural relationships in scenes using graph kernels. *ACM Trans. Graph.* 30:34:1--34:12.

[10] Ferrari, V., Marín-Jiménez, M., Zisserman, A. (2009). 2D Human Pose Estimation in TV Shows, 128-147. *Berlin, Heidelberg: Springer-Verlag.*

[11] Comaniciu, D., Meer, P. (2002). Mean shift: A robust appriach towar feature space analysis. *IEEE Transactions on Patter Analysis and Machine Intelligence* .

[12] Klir, G. J., Yoan, B. (1995). Fuzzy Sets and Fuzzy Logic Theory and Applications. *Prentice Hall PTR.*

[13] Matsakis, P., Wendling, L. (1999). A new way to represent the relative position between areal objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 101(1):634--643.

[14] Palus, H., Bogdanski, M. (2003). Clustering techniques in colour image segmentation. *Artificial Intelligence Methodos* , AI-METH 2003 .

[15] Tabbone, S., Wendling, L., Tombre, K. (2003). Matching of graphical symbols in line drawing images using angular signature information. *International Journal on Document Analysis and Recognition.*

[16] Skubic, M., Blisard, S., Baliley, C., Adams, J., Matsakis, P. (2002). Qualitative analysis of sketched routed maps: Translating a sketch into linguistic descrptions. *IEEE Trans. on System, Man, and Cybernetics PartB.*

[17] Gudivada, V. N., Raghavan, V. V. (1995). Design and evaluation of algorithms for image retrieval by spatial similarity. *ACM Trans. on Information Systems* 13(2):115--144.

[18] Miyajima, K., Ralescu, A. (1994). Spatial-organization in 2d segmented images: Representation and recognition of primitive spatial relations. *Fuzzy Sets and Systems* 65(2-3):225--236.