On Approach to Vision Based Fire Detection Based on Type-2 Fuzzy Clustering

Ha Dai Duong, Dinh Dung Nguyen, Long Thanh Ngo, Dao Thanh Tinh
Department of Information Systems, Faculty of Information Technology
Le Quy Don Technical University, Hanoi, Vietnam
Email: duonghadai@yahoo.com, ngotlong@gmail.com, tinhdt@mta.edu.vn

***Abstract*—During the last ten years computer vision techniques have shown a great potential in solving the problem of automatic fire detection. Vision-based fire detection offers many advantages over the conventional methods that use smoke and heat detectors. This paper presents a novel approach for fire detection by modeling the structure of spatial of fire, this structure is considered in terms of the color intensity of fire pixels. Furthers the type-2 fuzzy clustering technique is applied to separate fire-color pixels into some clusters, then these clusters are used to model structure of fire. Experimental results show that our method is capable of detecting fire in early state of fire and weak light-intensity environment; and this method uses only information on a single image so it can be integrated into the surveillance system that used dynamic camera.**

***Keywords*-Vision based fire detection, type-2 fuzzy clustering, type-2 fuzzy sets.**

I. Introduction

In the last ten years a large number of vision based fire detection systems are introduced due to the rapid development in digital camera technology and advances in content-based video and image processing. Vision based systems generally make use of three characteristics of fire: color, motion and geometry. In Healey et al. [1], authors determinate fire rely on movement and color alone. Phillips et al. [2], used color predicate information and the temporal variation of a small subset of images to recognize fire in video sequences; a manually segmented fire set is used to train a system that recognizes fire like color pixels and the results is used to form a look-up table for the fire detection system. Liu and Ahuja [3], proposed a vision based fire detection algorithm based on spectral, spatial and temporal properties of fires; analysis of the temporal fire variation to allow fire flickering. Chen et al. utilized a change detection scheme to detect flicker in fire regions [4]; the moving objects were filtered with fire and smoke filter to raise an alarm for possible fire in video and they used a generic fire and smoke model to construct the corresponding filter. Toreyin et al. [5], proposed an algorithm which combines a generic RGB color model, motion information and Markov process enhance fire flicker analysis; in another proposal they presented a real-time algorithm for fire detection in video sequences [6], combine of motion and color clues with fire flicker analysis on a wavelet domain.

Type-2 fuzzy logics with the capability of handing the uncertainty are widely applied in many fields [13], [14], especially pattern recognition [15]. Clustering algorithms have developed for applications of pattern recognition in different approaches. The family of k-mean clustering algorithms has archived positive results with refinements such as identifying the number of clusters [18]. Type-2 fuzzy c-mean clustering [16], [17] is an extension of fuzzy c-mean algorithm with identifying FOU of fuzzifiers, resulting to handling uncertainty is better.

The paper deals with a refinement algorithm of type-2 fuzzy c-mean clustering with initial step is to find rough center of clustering and identifying the number of clusters. This paper also proposes a new approach to vision-based fire detection by using interval type-2 fuzzy sets in clustering algorithm for fire-color pixels in RGB color space. The model of fire is built on the results of clustering includes number of clusters, number of pixels, shape and location of each cluster. The proposed approach is experimented with two data-sets to show fire will be detected early for alarming in weak light-intensity environment.

II. Preliminaries

A. Type-2 Fuzzy Sets

A type-2 fuzzy set in X is $\tilde{A}$, and the membership grade of $x \in X$ in A is $\mu_{\tilde{A}}(x,u), u \in J_x \subseteq [0,1]$, which is a type-1 fuzzy set in [0, 1]. The elements of the domain of $\mu_{\tilde{A}}(x,u)$ are called primary memberships of x in $\tilde{A}$ and the memberships of the primary memberships in $\mu_{\tilde{A}}(x,u)$ are called secondary memberships of x in $\tilde{A}$.

Definition 2.1: A $type-2\ fuzzy\ set$, denoted $\tilde{A}$, is characterized by a type-2 membership function $\mu_{\tilde{A}}(x,u)$ where $x \in X$ and $u \in J_x \subseteq [0,1]$, i.e.,

$$\tilde{A} = \{((x,u), \mu_{\tilde{A}}(x,u)) | \forall x \in X, \forall u \in J_x \subseteq [0,1]\} \quad (1)$$

or

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x,u))/(x,u), J_x \subseteq [0,1] \quad (2)$$

in which $0 \leq \mu_{\tilde{A}}(x,u) \leq 1$.

At each value of x, say $x = x'$, the 2-D plane whose axes are u and $\mu_{\tilde{A}}(x',u)$ is called a $vertical\ slice$ of $\mu_{\tilde{A}}(x,u)$. A *secondary membership function* is a vertical

978-1-4577-1196-1/11/$26.00 ©2011 IEEE

slice of $\mu_{\tilde{A}}(x,u)$. It is $\mu_{\tilde{A}}(x=x',u)$ for $x \in X$ and $\forall u \in J_{x'} \subseteq [0,1]$, i.e.

$$\mu_{\tilde{A}}(x=x',u) \equiv \mu_{\tilde{A}}(x') = \int_{u \in J_{x'}} f_{x'}(u)/u, J_{x'} \subseteq [0,1] \quad (3)$$

in which $0 \leq f_{x'}(u) \leq 1$.

Type-2 fuzzy sets are called an interval type-2 fuzzy sets if the secondary membership function $f_{x'}(u) = 1 \; \forall u \in J_x$.

B. Interval Type-2 fuzzy Clustering

Interval of primary memberships for a pattern is computed with two fuzzifiers m_1 and m_2 which represent different degrees of membership, $\overline{u}_{ik}$ and $\underline{u}_{ik}$ are determined as follows:

$$\overline{u}_{ik} = \begin{cases} \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})^{2/(m_1-1)}} & if \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})} < \dfrac{1}{C} \\ \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})^{2/(m_2-1)}} & if \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})} \geq \dfrac{1}{C} \end{cases} \quad (4)$$

$$\underline{u}_{ik} = \begin{cases} \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})^{2/(m_1-1)}} & if \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})} \geq \dfrac{1}{C} \\ \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})^{2/(m_2-1)}} & if \dfrac{1}{\sum\limits_{j=1}^{C} (d_{ik}/d_{jk})} < \dfrac{1}{C} \end{cases} \quad (5)$$

in which $i = \overline{1,C}$, $k = \overline{1,N}$, C is number of clusters, d_{ij} is Euclide distance between two patterns in m-dimensions space, N is number of samples.

In general, computation of fuzzy memberships in interval type-2 fuzzy C-means algorithm is achieved by computing the relative distance among the patterns and cluster centroids. Hence, to define the interval of primary membership for a pattern, we define the lower and upper interval memberships using two different values of m.

In (4)-(5), m_1 and m_2 are fuzzifiers which represent different fuzzy degrees. We define the interval of a primary membership for a pattern, as the highest and lowest primary membership for a pattern. These values are denoted by upper and lower membership for a pattern, respectively. The uses of fuzzifiers which represent different fuzzy degrees give different objective functions to be minimized in interval type-2 fuzzy C-means algorithm as:

$$\begin{cases} J_{m_1}(U,v) = \sum\limits_{k=1}^{N} \sum\limits_{i=1}^{C} (u_{ik})^{m_1} d_{ik}^2 \\ J_{m_2}(U,v) = \sum\limits_{k=1}^{N} \sum\limits_{i=1}^{C} (u_{ik})^{m_2} d_{ik}^2 \end{cases} \quad (6)$$

in which $d_{ik} = d(x-k-v_i) = \| x_k - v_i \| = \left[\sum\limits_{j=1}^{d} (x_{kj} - vij) \right]^{1/2}$.

Because each pattern has a membership interval as the lower $\underline{u}$ and the upper $\overline{u}$, each centroid of a cluster is represented by the interval between V_L and V_R. To find V_L and V_R, an iterative algorithm is used [16]. Cluster centroid is computed as follows:

$$v_i = \sum_{k=1}^{n} (u_{ik})^m x_k / \sum_{k=1}^{n} (u_{ik})^m \quad (7)$$

in which $i = \overline{1,C}$.

Compute the mean of centroid v_j

$$v_j = (V_R + V_L)/2 \quad (8)$$

For memberships

$$u_j(x_i) = (u_j^R(x_i) + u_j^L(x_i))/2, j = 1, ..., C \quad (9)$$

With

$$u_j^L = \sum_{l=1}^{M} u_{jl}/M, u_{jl} = \begin{cases} \overline{u}_j & \text{if } x_{il} uses \overline{u}_j \text{ for } v_j^L \\ \underline{u}_j & otherwise \end{cases} \quad (10)$$

$$u_j^R = \sum_{l=1}^{M} u_{jl}/M, u_{jl} = \begin{cases} \overline{u}_j & \text{if } x_{il} uses \overline{u}_j \text{ for } v_j^R \\ \underline{u}_j & otherwise \end{cases} \quad (11)$$

If $u_j(x_i) > u_k(x_i)$ for $k = 1, ..., C$ and $j \neq k$ then x_i is assigned to cluster j.

III. Fire model

In this proposal, we consider the problem of fire detection in environments with weak light-intensity and in the early states of fire; in these conditions, the flame is small and brighter than the background. Our method uses the clustering technique described in section II.B with some modification for each single image (see IV.A). The results of clustering is then considered by the fire model for detecting fire. A fire-alarm is given when the alarm-raising.

The fire region in a single image can be modeled as follows: (i) It has a high contrast to its surroundings; (ii) It exhibits a structure of nested rings of colors, changing from white at the core to yellow, orange and red in the periphery.

A. Environment with weak light-intensity

As mentioned above, the scope of these problem is to solve fire detection in condition of weak light-intensity environments. Visually, The video can be classified according to levels of the ambient light: weak, moderate or strong. Environmental conditions of weak light can be seen in the video recording at night, in a closed room, in low light

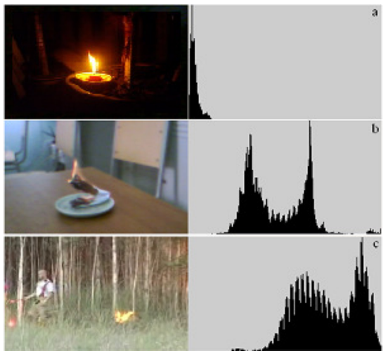

Figure 1. The images from the video frame and its gray-level histogram

.etc. These are fairly common situations in practice for problem of detection and fire observation. the environmental light conditions could be analysed by using a gray-level histogram. Figure 1 illustrates the histograms of the three images in different lighting conditions, (a) - Photos took in weak light-intensity environments, gray level histogram is located on the left, (b) - Photographs of the normal environment, the histogram is the gray balance in the horizontal axis, (c) - Photographs of outdoor lighting conditions to at sunny, histogram is located on the right.

Denote $p(r)$ be a normalized histogram of gray image f, we have:

$$p(r) = n_r/n \tag{12}$$

where $r \in [0.1, ..., L]$, and L is the largest gray level in images, n_r is the total number of pixels with gray level equal to r and n is the total number of pixels of the image f.

Easy to see that, in low light conditions, the average value of gray levels is small, and also image the surface structure will be relatively homogeneous. The average value calculated by:

$$M = \sum_{r=0}^{L} r * p(r) \tag{13}$$

and the uniformity on the image by:

$$Un = \sum_{r=0}^{L} p(r) * p(r) \tag{14}$$

Based on the parameters M and U_n to conclude image f was taken in low light conditions if:

$$M \leq M_0 \text{ and } Un \geq Un_0 \tag{15}$$

where M_0 and U_0 are the values of a predetermined threshold. Suggested experimental values $M_0 \in [80, 90]$ and $Un_0 \in [0.05; 0.07]$ for the relatively good results. Experiments in this paper use the value of $M_0 = 85$, and $Un_0 = 0.06$.

B. Spatial structure of fire

After clustering the fire, we generally have a spatial structure as Figure 2. The structure of fire in this proposal is a combination of clusters and their number of points, position and distribution of each class around their center.

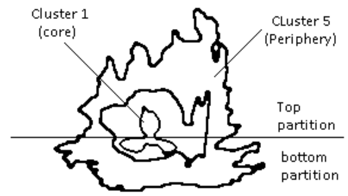

Figure 2. The spatial structure of clusters

1) Clusters: For each pixel in image, three values of RED, GREEN and BLUE channels are used to cluster. We cluster all pixels of the image in a RGB space into six clusters. The dark cluster is ignored because it is the background of image; five other clusters may belong to the fire blob, these clusters are numbered from 1 to 5 respectively as illustrated in Figure 2.

Approximately, probability of each pixel in fire blob belong to cluster i is as:

$$c_i = n_i/n \tag{16}$$

where $i \in \{1, 2, 3, 4, 5\}$, n_i is the total number of pixels of cluster i^{th} and n is the total number of pixels of fire blob.

The statistical analysis of probability of five clusters in fire pixels over a large set of images is performed. For this purpose a set which consists 765 images at different resolutions are collected from internet. The collected set of images has a wide range of illumination and camera effects. The result of analysis is reported in Table I.

Table I
PROBABILITY OF FIVE CLUSTER PIXELS OF FIRE

Cluster	C_1	C_2	C_3	C_4	C_5
$c_i^l - c_i^h$	0.01-0.17	0.02-0.22	0.04-0.26	0.11-0.37	0.27-0.57

Then, the considered single image f contains a fire pattern if the probability of each of clusters satisfies:

$$c_i^l \leq c_i \leq c_i^h \tag{17}$$

in which $i = \overline{1, 5}$.

2) Position of clusters in the image space: The above described experiments are implemented to get relationship of position between cluster C_1 and the remaining clusters in the image space as follows: to translate the cluster i into (x_1, y_1) by $(x_i = x_i - x_1, y_i = y_i - y_1)$; and normalization the distance from each cluster i and (x_1, y_1) by $(x_i = x_i/M, y_i = y_i/M)$, where M=max($\{x_i\}, \{y_i\}$) and $i = 1, 2, 3, 4, 5$. The results are shown in Figure 3.

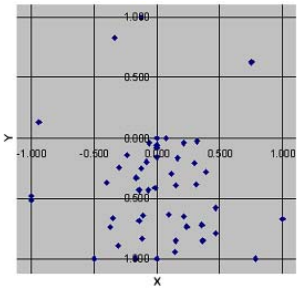

Figure 3. The position of the clusters

It is easy to recognize that most centroids of clusters are located in the rectangle by $x = [-0.5, 0.5]$ and $y = [0, -1]$. Additional criteria for fire in single image f if:

$$-0.5 \leq x_i \leq 0.5 \text{ and } 0.0 \leq y_i \leq -1.0 \tag{18}$$

with (x_i, y_i) is the location of center of cluster C_i.

3) Distribution of a cluster around its centroids: The last feature of fire is symmetric to the center of fire. We model this feature by dividing the image space into two partitions as in Figure 2: Top and Bottom partitions. Call N_i^T and N_i^B are numbers of pixels of fire blob in the top and bottom partitions of ith cluster; consider the ratio of Top-Cluster symmetry s_i defined by $s_i = N_i^T/(N_i^T + N_i^B)$. Do the same method has been used in section 1). The results of analysis are shown in Table II.

Table II
THE RATIO OF TOP-CLUSTER SYMMETRY

Cluster	C_1	C_2	C_3	C_4	C_5
$s_i^l - s_i^h$	0.06-0.61	0.12-0.68	0.03-0.71	0.06-0.65	0.12-0.68

Then, another additional condition to prove the single image f that contains fire pattern if:

$$s_i^l \leq s_i \leq s_i^h \tag{19}$$

Finally, the image f with clusters $C_i, i = \overline{1,5}$ results in fire alarm if these equations (17),(18) and (19) are satisfied.

IV. ALGORITHM

A. Modified Interval Type-2 Fuzzy C Means Clustering

In this section, we represent some modifications of interval type-2 fuzzy C Means clustering by computing an interval of primary memberships for a pattern with two fuzzifiers m_1 and m_2. We define $I_k = \{i | 1 \leq i \leq C, d_{ik} = 0\}$ in which $k = \overline{1, N}$ and d_{ik} is the Euclide distance between two patterns in m-dimensions space.

In case of $I_k = \emptyset$, $\overline{u}_{ik}$ and $\underline{u}_{ik}$ are determined as equations (4)-(4). Otherwise, if $I_k \neq \emptyset$, $\overline{u}_{ik}$ and $\underline{u}_{ik}$ are determined as follows:

$$\overline{u}_{ik} = \begin{cases} 0 & \text{if } i \notin I_k \\ \sum\limits_{i \in I_k} \overline{u}_{ik} = 1 & \text{if } i \in I_k \end{cases} \tag{20}$$

$$\underline{u}_{ik} = \begin{cases} 0 & \text{if } i \notin I_k \\ \sum\limits_{i \in I_k} \underline{u}_{ik} = 1 & \text{if } i \in I_k \end{cases} \tag{21}$$

in which $i = \overline{1, C}$, $k = \overline{1, N}$.

In the fuzzy C-Means algorithms, problem is how to initialize the centroid matrix V, because initialization of matrix V can cause affect on the number of steps and results of clustering. Hence, a method to generate the matrix V to make fuzzy C Means algorithms stable and efficient, called Initialization Centroid Algorithm, is shown to initialize centroid matrix V based on density of patterns. The centroid will be in the samples that the density surrounding the sample data are large. This step is quite large to affect the calculation process. The concept of statistical variance mathematical model is used to solve the problem of selecting a surrounding data points.

Initialization centroid Algorithm Consider $X = (x_1..x_N)$, $x \in R^M$ Step 1: Compute the expected pattern $\overline{z}_i = \frac{1}{N}\sum\limits_{j=1}^{N} x_{ji}$ and standard deviation $s_i = \sqrt{\frac{1}{N}\sum\limits_{j=1}^{N}(x_{ji} - \overline{z}_i)^2}$. with $i = 1, 2, ..., M$.

Consider the surround of each data point is M-dimensional box with radius can be defined by the standard deviation is $r = \min\limits_{1<i<M} s_i$.

Step 2: Compute density D_i of pattern x_i.

$$D_i = \sum_{j=1}^{N} T(r - \|x_j - x_i\|) \tag{22}$$

in which $u_{jl} = \begin{cases} 1 & \text{if } z \geq 0 \\ 0 & otherwise \end{cases}$

Step 3: Find pattern x_i with $D_i = max_{1 \leq j \leq N} D_j$
$V_u = V_u \cup x_i$
Step 4: Compute $CC = \{x_j | r_i - \|x_i - x_j\| \geq 0\}$
$X = X \ CC$
Step 5: If $X = \emptyset$ then go to Step 6 else back to Step 2.
Step 6: Given a set of candidate points V_u.

If V_u is large then we can proceed with this algorithm to reduce the number of candidate clusters.

We can speed up calculations by dividing the input data set into subsets, then proceed to apply the algorithm for that subset, we have candidates set V_i . Then we proceed with the candidate set $\cup V_i = V$, then apply this algorithm to the set V. Depending on the particular problem that we can apply the following measures to reduce the number of candidate clusters such as the shape of the clusters that we can remove the candidates in a straight line, or by ellipses ...Finally, the centroid matrix V can be initialized by choosing the patterns in V_u according to the density of candidates.

In addition to problems in getting the initial cluster, the fuzzy C-Means algorithms also have difficulty in determining exactly the number of clusters, depending on the problem that we can choose the number of clusters different.

Commented that the process of clustering in many cases there are very little sample data clusters, even without sample data. Thus depending on the requirements of the problem we can remove empty clusters. Therefore, we propose algorithms to adjust the cluster, eliminating unnecessary clusters based on density and size of the clusters. The number of clusters c is input for clustering process, the small clusters will be attracted to large clusters and would be eliminated. The resulting output is $c^*(c^* < c)$ clusters. Based on the idea in [18], the clustering based on density will result in to discover clusters with arbitrary shape. The following is a modified Interval Type-2 Fuzzy C Means Algorithm to automatically adjust the number of clusters.

Modified Interval Type-2 Fuzzy C Means Algorithm (IT2-FCM)

Algorithm determines the actual cluster c^* is divided into two phases. In the first one, Interval Type-2 fuzzy clustering is used for clustering. Assume that the number of clusters c is larger than the number of clusters c^* in reality.

Step 1: Initialize centroid matrix V: $V = [v_{ij}], V^{(0)} \in R^{M\times C}, j = 0$ using *the initialization centroid algorithm.*

Step 2: Take a random sample of data x_t from the preliminary data for $j = 1, 2, ..., C$. Calculate the membership function according to the formulas (20)-(21) and (9).

Step 3: Update centroid of clusters $[V = [v_1, v_2, ..., v_c]$ according to (8) and result of step 2: $U^{(j)}$.
Repeat step 2 and step 3 until all clusters without changing. However, depending on the particular problem we will determine the rest of this loop, when the cluster to change the a certain threshold value or simply specify the maximum number of loops.

In the second phase, loop includes two main steps: Step 1: For each sample data x_t, calculate membership function according to formulas (20)-(21). A new distance is defined which depends on the density of the cluster.

$$dm(x_t, C_i) = |u_{ti} - E_i \log_2(p(C_i)) \tag{23}$$

$p(C_i)$ can be determined $p(C_i) = I(C_i)/N$

In which $I(C_i)$ is the number of sample data of the cluster i^{th} and N is the total number of input sample data. Parameter E_i in 23 is calculated by the formula $E_i = E/r_i$, in which r_i is ith cluster radius and E is a constant to adjust the level of density to influence the clustering results. Through experimental results, E is often chose in the range $E \in (a, 3a]$, where $a = average(r) + average(d/2)$, r is the radius of the cluster and output of the first phase, d is the smallest distance between two clusters that are larger than $3r$ [18].

Next, matrix $U^{(j)}$ is computed by (20)-(21) and (9) with $d_{ij} = dm_{ij}$

Step 2: Update centroid of clusters $[V = [v_1, v_2, ..., v_c]$.

Figure 4. Example on clustering to get spatial structure of fire

Step 3: Remove redundant clusters (empty clusters). Repeat step 1, 2 and 3 until all clusters without changing to get number of clusters $c^*(c^* \leq c)$.

B. Complete Fire Detection Algorithm

The completed algorithm is a combination of the method of clustering fire blob in a single image f and the checking of fire model as described above. The condition of environment is ignored in this algorithm, it is added if necessary.

Input: Image f.

Output: $A = TRUE$ if f has fire, $A = FALSE$ if it has not.

Algorithm is performed in two phases.

In phases one: Perform clustering f into 2 clusters by modified IT2-FCM. End of this phases we relatively obtain a suspected area $f*$ and a environment area

In phases two:

Step 0: $A = FALSE$;

Step 1: Check whether $f*$ is taken in weak-light intensity environment by (15); if ok do step 2, otherwise do step 5;

Step 2: Cluster $f*$ into 5 clusters C_1, C_2, C_3, C_4, C_5 by IT2-FCM with The binding of centroid initialization ;

Step 3: If the $C_i, i = \overline{1, 5}$ are satisfied (17),(18) and (19) then do step 4; else go to step 5;

Step 4: $A = TRUE$;

Step 5: Return A;

V. Experimental results

We use a large collection of 200 images obtained from Internet. However, this collection is divided into sub-sets: i) Set A: images obtained from environments with weak light intensity; ii) Set B: images without fire. The modified interval type-2 fuzzy c-mean clustering is used to get spatial structure of fire images. The Figure 4 is example on clustering to get spatial structure of fire. Our testing results on these sets for fire detection were reported in Table III.

Table III
SUMMARY OF TEST RESULTS

Catalog	Number image	True Alarm	False Alarm
Set A	100	96	4
Set B	100	93	7

In general, we observe that the proposed method give the best performance on images taking from environments

Figure 5. Images consist fire are taken in weak light conditions

with weak light-intensity or without fire (Set A or Set B). We will take a closer look at Set A in order to obtain an understanding about the behavior of our method. The results on Set A, as reported in Table III, indicate that our method obtained a high percentage of true alarms (96%). However, there were still several cases where our method wrongly detected fire. In Figure 5, we demonstrate three typical cases with three images from Set A where our method is either successful or failed (original images are on the left column, while the right column is for clustered images). The first image is classified as containing fire pattern, in which five clusters are found and are consistent with the proposed model. Meanwhile, the test on the second image give a wrong answerer. In this case, the fire is too small to be represented by our proposed model (clusters found from $C1$ to $C5$ are: 0.5, 0.0, 0.1, 0.1 and 0.3). The final image is also the case our method failed to detect fire. The reason is that the fire area is too large, that made clustering being not suitable for our method.

VI. Conclusions

In this paper, an approach to vision based fire detection is developed; the main idea of this proposal is modeling of fire in terms of clusters. Authors use interval type-2 fuzzy C mean algorithm with some modification. This approach can be used for detection of fire in video databases, as well as real-time detection of fire. The experimental results show a good performance of our method in some cases such as images obtained from environments with weak light-intensity. For the future works, we intend to incorporate a mechanism to deal with semantic features of fire.

Acknowledgment

This work was funded by the Vietnam National Foundation for Science and Technology Development (NAFOSTED), Grant No 102.012010.12. and the Reseach Fund RFit@LQDTU, Faculty of Information Technology, Le Quy Don Technical University.

References

[1] G. Healey, D. Slater, T. Lin, B. Drda, A.D. Goedeke, *A system for real time fire detection* , Proc. IEEE Computer Vision and Pattern Recognition Conf. (CVPR- 93), pp. 605-606, 1993.

[2] W. Phillips, M. Shah, N.V. Lobo, *Flame recognition in video*, Pattern Recognition Letter, Vol 23, pp. 319-327, 2002.

[3] C. Liu, N. Ahuja, *Vision based fire detection*, Proceedings of the 17th Int. IEEE Conf. on Pattern Recognition, 2004.

[4] T. Chen, P. Wu, Y. Chiou, *An early fire-detection method based on image processing*, Proc. IEEE Internat. Conf. on Image Processing, ICIP 04, pp. 1707-1710, 2004.

[5] B.U. Toreyin, Y. Dedeoglu, and A.E. Cetin, *Flame detection in video using hidden Markov models*, Proc. IEEE International Conference on Image Processing, pp. 1230-1233, 2005.

[6] B.U. Toreyin, Y. Dedeoglu, U. Gudukbay, and A.E. Cetin, *Computer vision based method for real-time fire and flame detection*, Pattern Recognition Let., Vol 27(1), 49-58, 2006.

[7] Ha Dai Duong, Nguyen Anh Tuan, *Using Bayes method and Fuzzy C - Mean Algorithm for Fire Detection in Video*, Proc. The 2009 Int. Conf. on Advanced Technologies for Communications, 141-144 ,Haiphong, Vietnam, 2009.

[8] Ha Dai Duong, Dao Thanh Tinh, *Fire detection in environment with weak-light intensity*, Proc. The National Conference on Selected Issues of Information Technology and Communications, HungYen, Vietnam, 2010.

[9] D. V. Prokhorov, *Adaptive Critic Designs and Their Applications* , Ph.D. Dissertation, Texas Tech University,1997.

[10] J. Si, A. Barto, W. Powell and D. Wunsch (Editors), *Handbook of Learning and Approximate Dynamic Programming*, New York: IEEE and Wiley, 2004.

[11] P. J. Werbos, *Approximate dynamic programming for real-time control and neural modeling Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches* (Chapter 13), Edited by D. A. White and D. A. Sofge, New York, NY: Van Nostrand Reinhold, 1992.

[12] P. J. Werbos, *Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research*, IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-17, pp. 7-20, Jan./Feb., 1987.

[13] N. Karnik and J.M. Mendel, *Operations on Type-2 Fuzzy Sets*, Fuzzy Sets and Systems, Vol. 122, pp.327-348, 2001.

[14] J.M. Mendel and R. I. John, *Type-2 Fuzzy Sets Made Simple*, IEEE Trans. on Fuzzy Systems, Vol. 10(2), pp. 117-127, 2002.

[15] J. Zeng and Z.Q. Liu, *Type-2 Fuzzy Sets for Pattern Classification: A Review*, Proceedings of the 2007 IEEE Symposium on Foundations of Computational Intelligence, FOCI07, pp.193-200, 2007.

[16] C. Hwang and F.C.H. Rhee, *Uncertain Fuzzy Clustering: Interval Type-2 Fuzzy Approach to C-Means*, IEEE Trans. on Fuzzy Systems, Vol 15(1), pp. 107-120, 2007.

[17] M.H. Zarandi, M. Zarinbal, I.B. Turksen, *Type-II Fuzzy Possibilistic C-Mean Clustering*, Proc. IFSA-EUSFLAT 09, 30-35, 2009.

[18] Krista Rizman Zalik, *An efficient k'-means clustering algorithm*, Pattern Recognition Let., Vol. 29, 1385-1391, 2008.