Object Modeling for Multicamera Correspondence Using Fuzzy Region Color Adjacency Graphs

Amir Hossein Khalili[1] and Shohreh Kasaei[2]

[1] Sharif University of Technology, Tehran, Iran
a_khalili@ce.sharif.edu

[2] Sharif University of Technology, Tehran, Iran
skasaei@sharif.edu

Abstract

In this paper, a novel moving object modeling suitable for multicamera correspondence is introduced. Taking into consideration the color and motion features of foreground objects in each independent video stream, our method segments the existing moving objects and constructs a graph-based structure to maintain the relational information of each segment. Using such graph structures reduces our correspondence problem to a subgraph optimal isomorphism problem. The proposed method is robust against various resolutions and orientations of objects at each view. Our system uses the fuzzy logic to employ a human-like color perception in its decision making stage in order to handle color inconstancy which is a common problem in multiview systems. The computational cost of the proposed method is made low to be applied in real-time applications. Also, it can solve the partial occlusion problem more precisely than the Meanshif occlusion solver by 15.7%.

Keywords

Object Correspondence, Multicamera Tracking, Region Adjacency Graph, Fuzzy Color Modeling.

1. Introduction

The increasing demand for analyzing moving objects behavior in wide area has heightened the need for robust modeling and correspondence of the moving objects in scene. The purpose of object modeling is taking out some features from images of an object so that the selected features are stable and reliable and could exactly discriminate the target in consequence projections. This task is difficult because an object in monitoring systems can move fast and unpredictably, can appear in a variety of poses and colors, and are often surrounded by clutter.

For applications such as outdoor video surveillance, where the entire scene is not coverable with a single camera, a distributed camera network should be implemented. In multiview framework, differences in camera directions, distances to target, illumination situations and quality of capturing cause single object is looked differently in each field of view. Changes in size, view direction, luminance and color values of objects are some artifacts which make the problem more challenging. For that reason, special object model is needed to maintain correspondence of two-dimensional projections of an object seen in different views.

Wide variety of methods has been reported to model moving objects in multiview applications, but a few of them handle color variation and occlusions effectively without employing high-level reasoning procedure and predefined model of target objects. In our experience the low level image features play a crucial role.

The proposed method for object modeling aims to segment moving object efficiently according to its color and motion features and accommodate human-like color perception to deal with uncertainty in observed colors. The proposed method constructs a region adjacency graph for each moving object perspective to book relational state of each segment and tries to reduce moving object correspondence problem to a subgraph optimal isomorphism problem. The proposed method doesn't need any high-level definition of moving objects and could deal with partial occlusions efficiently.

The rest of paper is organized as follows. In section 2 an overview on related previous works is given. In section 3 different steps of the proposed algorithm are introduced in detail. The experimental results are shown in section 4. Finally, section 5 concludes the paper.

2. Previous Work

Constructing object model has been extensively studied in image retrieval applications where the principle is extracting stable features for target object and looking for them in set of images. Most successful idea in this area employ time consuming algorithms using wavelet, Gabor filter or Hough transform [1,2,3], which are not applicable in real-time applications such as environment monitoring.

In context of tracking, objects are modeled differently. During tracking with single camera, objects could be modeled according to their temporal status information including position, temporal velocity, appearance features and color properties. These kind of modeling is robust and stable in environments included people and vehicles [4].

While tracking in multi cameras network, selected features should have little changes from different views, so that specific object could be discriminate easily from other cameras to track. Variation in cameras used in a network and their different positions, directions and distances to target, in addition to different illumination conditions cause selection of stable features in different views to be a challenging problem.

Most multicamera applications use color information of target objects as modeling parameters. [5,6] used average color of moving objects in HSV space. [7,8] use only one color histogram model per object. Although color histograms are rotation/scale invariant, computationally efficient and robust to partial occlusion, but prevent differentiating a person wearing blue jeans with a red shirt from a person wearing red pants and a blue shirt. Also, ordinary histogram matching approaches fail to discover uniqueness of sensed colors in multicamera application where a single color sensed in different value by common color spaces. Reference [9] tried to find and correct such distortion through a training step. Reference. [10] used Munsell color space [11] for constructing histograms. In this approach colors are coarsely quantized into 11 predefined bins. [12] used color-spatial distribution of moving object by partitioning moving blob in its polar representation. Some multicamera person tracking methods partition body to predefine number of segments and track color information of each independently. Reference [13] claims that color information of body, pelvis, and feet are the most stable information even in distributed cameras. In [14] authors cluster a person according to its color and motion features using watershed and K-means algorithm to recognize gestures and track each.

One approach to the detection and tracking problem is to fit explicit object models of shape, such as rigid wire-frame CAD models [15,16] or flexible active shape models [17]. Some model fitting approaches focused on high-level reasoning [18,19]. They predefine a model for their target objects, and try to fit objects' projections into it and estimate object's pose in other views. The robustness of such approaches is highly depends on the defined model. Models cover limited range of objects and behaviors and fails if the moving objects move fast or don't obey the definite behaviors. For such methods calibration data of cameras should be known. Occlusions, low resolution images and variety of poses and colors which are common in most applications are some crucial peril for robustness of these methods.

3. Proposed Method

The subsequent steps of the proposed moving object modeling for multiview correspondence are explained in detail in the following subsections.

3.1. Moving Object Segmentation

Background subtraction method [20] used for extracting foreground regions in each camera. Foreground regions are then modeled as region adjacency graph and are tracked independently in each frame of each view using method defined in [21].

Needing to be real-time and object independent, our segmentation method differs to what used in [14]. In our implementation two features used to segmentation foreground objects: color information and direction of motion vectors. In this regard we used Horn & Schunck algorithm [22] to compute optical flows and Cr and Cb channels of YCrCb color space as color feature.

The advantage of using motion vectors in segmentation processes is that the direction and distribution of the optical flow can be used to distinguish the different moving parts of objects. Even though these parts may have similar color in one view, they usually differ in others. By segmenting homogenous color parts according to their motion, no extra process is needed to split and merge different part of object model to match them in different views. In addition, motion is usually a reliable feature to detect human activity in indoor environment and from different outlooks [14].

Before segmentation a preprocessing is needed to smooth isolated pixels which their color or optical flow don't confirm with their neighbors. We suggest using bilateral smoothing filter which is presented in (1).

$$h(x) = \int_{\infty}^{\infty}\int_{\infty}^{\infty} f(\xi)C(\xi - X)S(f(\xi) - f(x))d\xi \quad (1)$$

In which both the closeness function, C, and the similarity function, S, are Gaussian functions of the Euclidean distance between their arguments. It is a simple, non-iterative scheme for edge-preserving smoothing. It replaces the pixel value at x with an average of similar and nearby pixel values and doesn't introduce false colors around the boundaries. By implementing such filter 3-5 times, we obtain graphic like appearance in which fine texture has gone.

The desired segmentation algorithm must have low complexity and strong power of discrimination. The choices in this regard are quite vast. We adopt the approach introduced in [21] because of three major reasons: 1) in that method it has been tried to act upon global characteristics of pixels. 2) It has low computational complexity and can be applicable in nearly linear time. 3) The graph-based structure of this algorithm highly conforms to our region adjacency graph (RAG) model. In this sense root nodes of segmentation algorithm which represent different regions correspond to nodes of RAGs. Two nodes in RAG are adjacent if their corresponding nodes in segmentation algorithm are adjacent in more than predefined threshold pixels. Fig. 1 shows an artificial image and its RAG yielded by color segmentation. Fig. 2 shows results of our proposed foreground segmentation method in a natural scene and its constructed RAGs.

When there is no high-level description of moving objects, RAG modeling gives useful structure to describe moving object and its parts. For the purpose of correspondence in different views some reliable and stable features of regions should be associated to each node of RAG. Color of regions is usually used as a robust feature in such applications. However, common color spaces do not present reliable and stable feature. In the next section our human perception-based color modeling is introduced to solve this shortcoming.

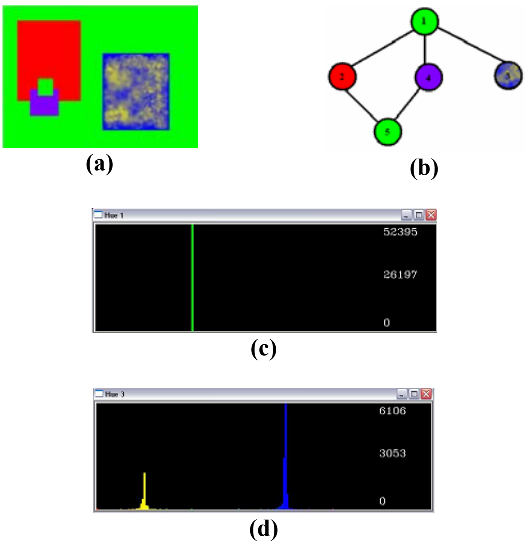


Fig. 1. a) A sample synthetic image. b) Region adjacency graph produced from color segmentation using [21]. c) Hue histogram of Region 1. d) Hue histogram of Region 3.

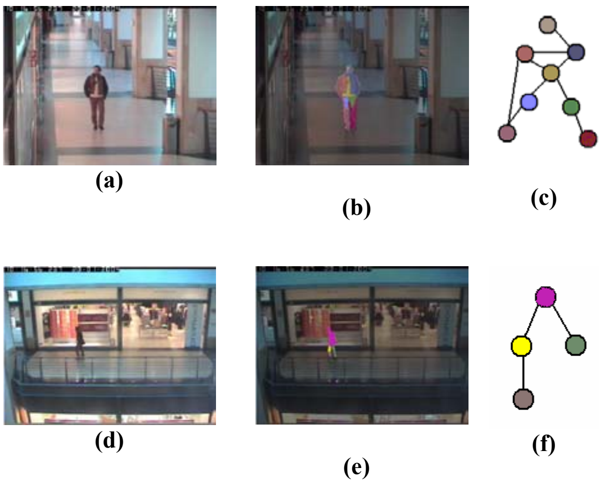


Fig. 2. Result of proposed segmentation in a natural scene. a) Original corridor frame. b) Foreground segmentation of a. c) Region adjacency graph of b. d) Original frame of front view. e) Foreground segmentation of d. f) Region adjacency graph of e.

3.2. Human Perception-Based Color Modeling

Color features are one of available features in multicamera tracking applications. Difference in view directions, target distances, illumination situations and capturing qualities causes single color of object is sensed dissimilar in each view to the extend that human faces a dilemma to recognize the object. Using common color spaces, the values which are assigned to the sensed colors differ such that common similarity measurement functions found them unlike and mismatch them incorrectly.

In our experience, in multiview applications human classifies colors to a limited set which is more stable in different views. Mapping colors of object projection in a view to members of this set, human looks for any arrangement of colors in other views that confirms with his target model.

Human color classification is not course. He some times faces a dilemma about a color and cannot determine to which member the specific color exactly belongs. In such a situation human assigns a membership value to each member to which the color is similar. Course partitioning the color space into a relatively small number of course segments such as [11] does not provide an optimal solution for correspondence problem.

To help our system to have humanoid color classification, we used HSL color space and fuzzy logic systems, and provide fast color segmentation using linguistic rules of human intuition. HSL is a color space which is more intuitive and closer to the human perception of color. Fuzzy logic is used as an interface between logic and human perception.

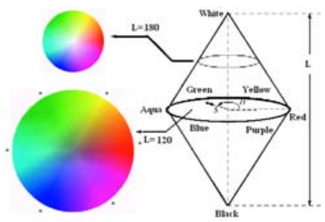
Since in HSL color space each color is defined by three values (H,S and L), the fuzzy logic model has three antecedent variable (*Hue, Saturation* and *Luminance*) and one consequent variable, which is a linguistic color.

The fuzzy sets of the antecedent fuzzy variable *Hue* are defined based on 6 basic hues which are distinguishable by human perception. We can see such classification in identification of different colors of rainbow. As shown in Fig. 3.a, the basic hues are *Red*, *Yellow*, *Green*, *Aqua*, *Blue*, *Purple*. The membership function of each is shown in Fog. 3.b distributed in 0-180.

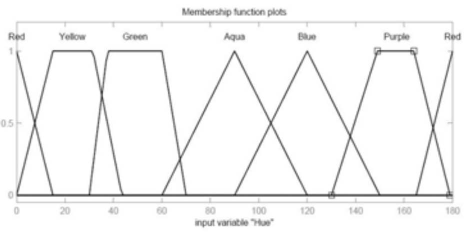
Saturation is defined using three fuzzy sets *Gray*, *Medium*, *Clear*, as shown in Fig. 4.

Luminance is also defined using the three fuzzy sets *Dark*, *Medium* and *Bright* as described in Fig. 5.

The consequent part of each fuzzy rule is a crisp discrete value of the set *Black*, *White*, *Red*, *Orange*, *Yellow*, *Gray*, *Brown*, *Aqua*, *Blue*, *Green*, *and Purple*. The members of this set are called *Linguistic* colors.



(a)



(b)

Fig. 3. a) HSL color space. 6 basic colors distinguishable with human perception are denoted by the "*" sign. b) Experimental membership functions of basic *Hue*s.

By defining set of fuzzy rules according to human belief and observation, perception of linguistics colors based on antecedent variables (*Hue*, *Saturation* and *Luminance*) are translated to system, so that system could inference the linguistic colors and their membership believes similar to the human. For example, the rule "$Yellow \wedge Medium \wedge Medium \rightarrow Brown$" is defined by manually classifying the color produced by the HSL triple such that the values of *H*, *S* and *V* are the points of maximum of the membership functions associated with the fuzzy sets *Yellow*, *Medium* and *Medium*. Based on the membership functions described in Fig. 3.a, 4 and 5, in this case the values are H = 25, S = 128, L =150. As another example the rule "$Blue \wedge Medium \wedge Bright \rightarrow Aqua$" is the rule which is activated mostly when H = 120 S = 125, L = 220. The color produced by this HSL triple would be classified by most human observers as *Aqua*. This corresponds to the natural language human perception-based rule "if the hue is *Blue*, the saturation is *Medium* and the value is *Bright* then the color is "*Aqua*". Thus, the set of fuzzy rules includes rules such as:

$Red \wedge Gray \wedge Bright \rightarrow White$

$Red \wedge Medium \wedge Dark \rightarrow Black$

$Red \wedge Gray \wedge Dark \rightarrow Black$

$Red \wedge Gray \wedge Medium \rightarrow Gray$

$Yellow \wedge Clear \wedge dark \rightarrow Orange$

$Red \wedge Clear \wedge Medium \rightarrow Red$

$Blue \wedge Medium \wedge Bright \rightarrow Aqua$

$Aqua \wedge Clear \wedge Medium \rightarrow Aqua$

Since the model has 6 fuzzy set for Hue, 3 for Saturation and 3 for luminance, the total number of rules required for this model is $6 \times 3 \times 3 = 54$. The reasoning procedure is based on a zero-order Takagi-Sugeno model.

In proposed method, base on H, S, L values of a pixel membership beliefs to each linguistic color is computed. Accumulative beliefs of region's pixels to each 11 linguistic colors, construct a histogram. After dividing each cell of the histogram to area of the region, the it is called *linguistic color histogram.* For each node of RAG a linguistic color histogram is associated.

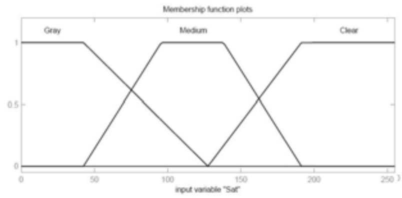


Fig. 4. Membership functions of *saturation* input variable. Membership functions have a great overlap due to the fact that saturation changes extremely between views.

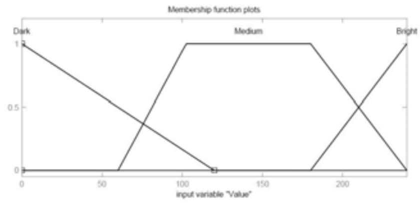


Fig. 5. Membership functions of *luminance* input variable. Membership functions have a great overlap due to the fact that saturation changes extremely between views.

3.3. Matching Two Fuzzy Region Adjacency Graphs

Similarity between two objects modeled in term of RAGS can be approximated by measuring subgraph optimal isomorphism of their graphs. Formally, two graphs are said to be isomorphic, if there is a one to one correspondence between their vertices and edges such that incidence relationship is preserved. If the isomorphism is encountered between a graph and a subgraph of another larger graph, then the problem is called subgraph isomorphism. When nodes of graphs are attributed and perfect matching between attributed values of graphs is target the problem is called subgraph optimal isomorphism.

Due to changes in direction, size and colors of projections of an object, we cannot expect to have graph isomorphism between viewed RAGs of a single object. Therefore it is more robust to use subgraph optimal isomorphism as similarity measurement. Also, thanks to subgraph optimal isomorphism we can discriminate RAGs of different objects in case of partial occlusion.

The main drawback of subgraph optimal isomorphism lies in its inherent computational complexity. The subgraph isomorphism problem is known to be NP-complete, even for planar graphs. Many efforts have been directed to find efficient algorithm for this purpose. We approximate this similarity measurement by adapting method introduced in [23]. This method uses minimum cost sequence of elementary graph manipulation operators as similarity measurement between two graphs. These operators which transform one graph, say *v*, into the another graph, say *h,* consist of: 1) deleting a node or an edge from v, 2) inserting a node or an edge into v, 3) substituting a node from v by a node from h.

According to [23] it is better to decompose the two candidate RAGs to smaller subgraphs called Basic Attributed Relational Graphs (BARG). A BARG is one level tree, including a node (denoted by n) and all its neighbors. Fig. 6 shows different BARGs of presented RAGs in Fig 2.d.

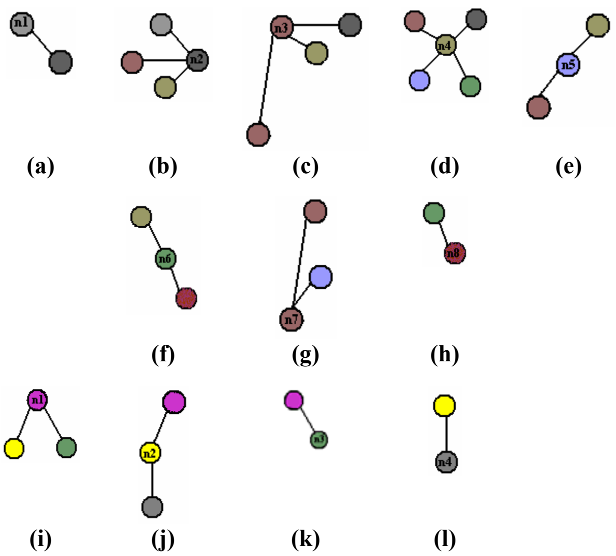


Fig. 6. (a-h) Basic attributed graphs of region adjacency graph of Fig. 2.c. (i-l) Basic attributed graphs of region adjacency graph of Fig. 2.e.

Subgraph optimal isomorphism measurement can be calculated with optimal matching of a complete weighted bipartite graph whose nodes are BARGs of each RAG partitioned in two groups. Weight of each connecting edge in bipartite graph is similarity distance between BARGs at two tails of the edge. Similarity distance of i'th BARG of RAG v in camera C1, to j'th BARG of RAG u in camera C2, is computed using (2).

$$\begin{aligned}\underset{c1,c2}{Dist}(v_i,u_j) &= \underset{c1,c2}{Dist}(n_{v_i},n_{u_j}) \\ &+ w_n \times \left|\deg(n_{v_i}) - \deg(n_{u_j})\right| \\ &+ w_e \times \left|\deg(n_{v_i}) - \deg(n_{u_j})\right| \\ &+ \underset{c1,c2}{Dist_{bipartite}}(N(n_{v_i}),N(n_{u_j}))\end{aligned} \tag{2}$$

in which $Dist(n_{v_i},n_{u_j})$ denotes similarity measurement between root nodes of two BARGs which is computed using adapted accumulative fuzzy intersection formula (3).

$$\underset{c1,c2}{Dist}(n_{v_i},n_{u_j}) = \sum_{y,x\in\ linguisti\ colors} (w_{x,y}^{c1,c2} \times \min(n_{v_i}.hist(x),n_{u_j}.hist(x)))^2 \tag{3}$$

$w_{x,y}^{c1,c2}$ is a coefficient which introduces similarity between two linguistic colors in two cameras. For example It is likely to visit a gray color in one view as black one in another view. So we define $w_{gray,black} = 0.5$. Similarity of black and white is very low in compare with similarity of black and gray, therefore we have $w_{white,black} << 0.5$. In other words, $w_{x,y}^{c1,c2}$ can be seen as the substituting cost of two nodes. We can define these coefficients manually according to color digression seen in two cameras, c1 and c2. Also, it is possible to obtain the coefficients according to seen colors of definite moving object in a training step. Finding two regions are correspondence, the $w_{x,y}$ value of their major linguistic colors are increased by α factor. $w_{x,y}$ s are updated independently for each perspective image of an object in each two cameras.

When two regions have great number of pixels in common at specific linguistic color, formula (3) increases their similarity measurement more significant than status in which they have same amount of intersection but distributed in more linguistic colors.

$\deg(n_{v_i})$ in (2) denotes degree of root of ith BARG of RAG v. w_n and w_e denote node and edge insertion costs. In our experience it is better to define w_n and w_e adaptive and as inverse function of difference in size of foreground objects. If two candidate foreground objects are in same size, w_n and w_e are assigned to their maximum values. Increasing in size difference of two foreground objects, causes decrease in amount of these coefficients. In this manner we can efficiently handle different resolutions of object in different cameras. In case of occlusion, w_n and w_e don't change and their last amounts are used. Thus, it is necessary to archive amounts of these variables for each BARG of each two candidate RAGs.

$Dist_{bipartite}(N(n_{v_i}),N(n_{u_j}))$ is calculated as the maximum matching of a complete weighted bipartite graph constructed from neighbors of n_{v_i} and n_{u_j}. Neighbors of n_{v_i}, $N(n_{v_i})$, are partitioned in one hand and neighbors of n_{u_j}, $N(n_{u_j})$, on the other hand. The weight of connecting edges is computed using (3). Fig. 7 shows such bipartite graph for BARG c and BARG j of

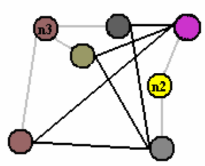


Fig .7. Bipartite graph constructed from neighbors of root of c and j BARGs of Fig .6. Root nodes and all their connecting edges do not participate in bipartite graph and are only demonstrated for transparency.

Fig. 6.

3.4. Tracking Using Region Adjacency Graphs

Fig .8 demonstrates an overview of our multiview tracking model.

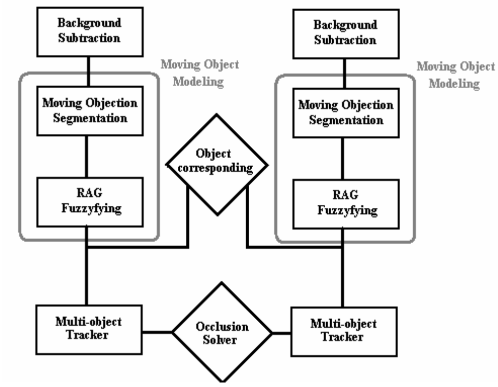


Fig .8 Block diagram of proposed multiview tracker.

Using methods introduced in sections 3.1 and 3.2, fuzzy region adjacency graph for each foreground object of each camera is produced. For tracking objects in each camera we use combination of sequential Kalman filter and joint probabilistic data association (JPDA) [25]. Similarity measurement introduced in section 3.3 is used to find most feasible hypothesis in validation gate of JPDA. For tracking in single camera we set $w_{x,y} = 1$ iff $x = y$ and $w_{x,y} = 0$ if $x \neq y$. It is due to the fact that we have not color distortion in a single view. Best matching consequence between RAGs of two views is used to define correspondence between objects. Finding two moving regions are correspondence, $w_{x,y}$ of their major linguistic colors are updated, section 3.3.

In case of occlusion, RAG of foreground in place of occlusion is called Super Region Adjacency Graph (SRAG). Tracker could segment occluding objects from occlusion area by examining subgraph isomorphism of previous RAG of each participating objects. The RAG which can be segmented completely from SRAG is RAG of most closed objects. By removing founded RAGs from SARG we can do subgraph isomorphism process again to locate other objects. The projection of object's bounding box from other view to considering view, limits our search through SARG.

4. Experimental Results

To evaluate usefulness of our proposed method, we implemented the multiobject tracking system in C++ using OpenCV infrastructure. We run our code on CPU 3.00 GHZ with 2.00 Gigabyte RAM. Fuzzy rules are constructed in Matlab environment and its produced .fis file is loaded in our program.

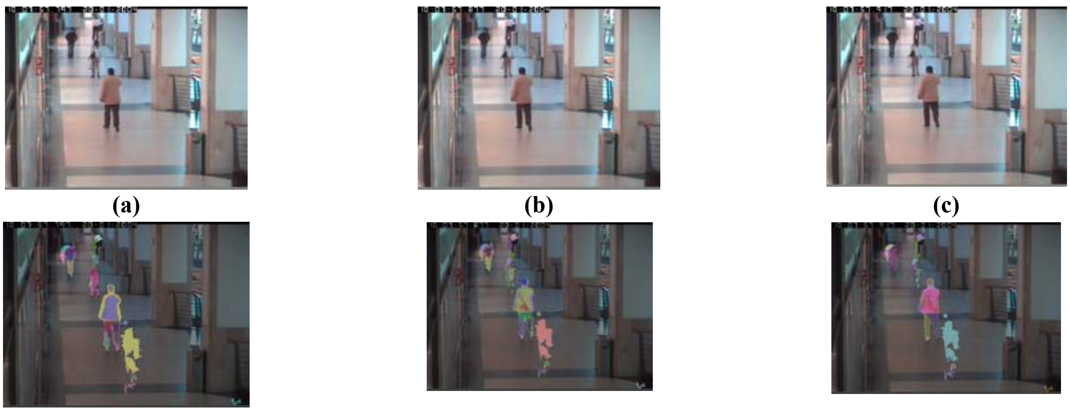
We have used outdoor natural scenes, which are considered to be more challenging for color segmentation and correspondence methods. For this purpose, we used the CAVIAR[25] and PETS2001[26] as our datasets.

CAVIAR consists of sequences across the hallway in a shopping centre in Lisbon.. For each sequence, there are two synchronized orthogonal videos, one with the view across the hallway and the other along it. The format is half-resolution PAL standard (384×288 pixels, 25 frames per second).

PET2001 consists of outdoor sequences taken from a university location. Each dataset includes two videos viewing environments each of which captured with 30 frames per second and frame size 768×520. We resized each frame by half to increase our processing speed. According to the distance among the location of moving objects and cameras, the captured videos mostly provide low resolution objects. This in fact causes a major problem for most available methods.

4.1. Moving Object Segmentation

Fig. 9 shows three consequent frames of "TwoEnterShop1" sequence of CAVIAR captured along a corridor view. It is noticeable how well different parts of the frontier person are segmented (head, hands, body, legs and feet are segmented efficiently). However, by going far and thus decreasing in resolution of moving objects (*e.g.,* far persons in Fig. 9) the accuracy is reduced, but we still obtain meaningful segments. The abnormally segments in the middle of these frames are due to false background segmentation. Fig.10 shows our results obtained from four pair frames of "OneStopNoEnter2" sequences in CAVIAR. Fig. 2



(a) (b) (c)

(d) (e) (f)

Fig .9. Three consequent frames of "*TwoEnterShop1*" sequence of CAVIAR along the corridor view. Colors associated to each segment in each frame are chosen in random.

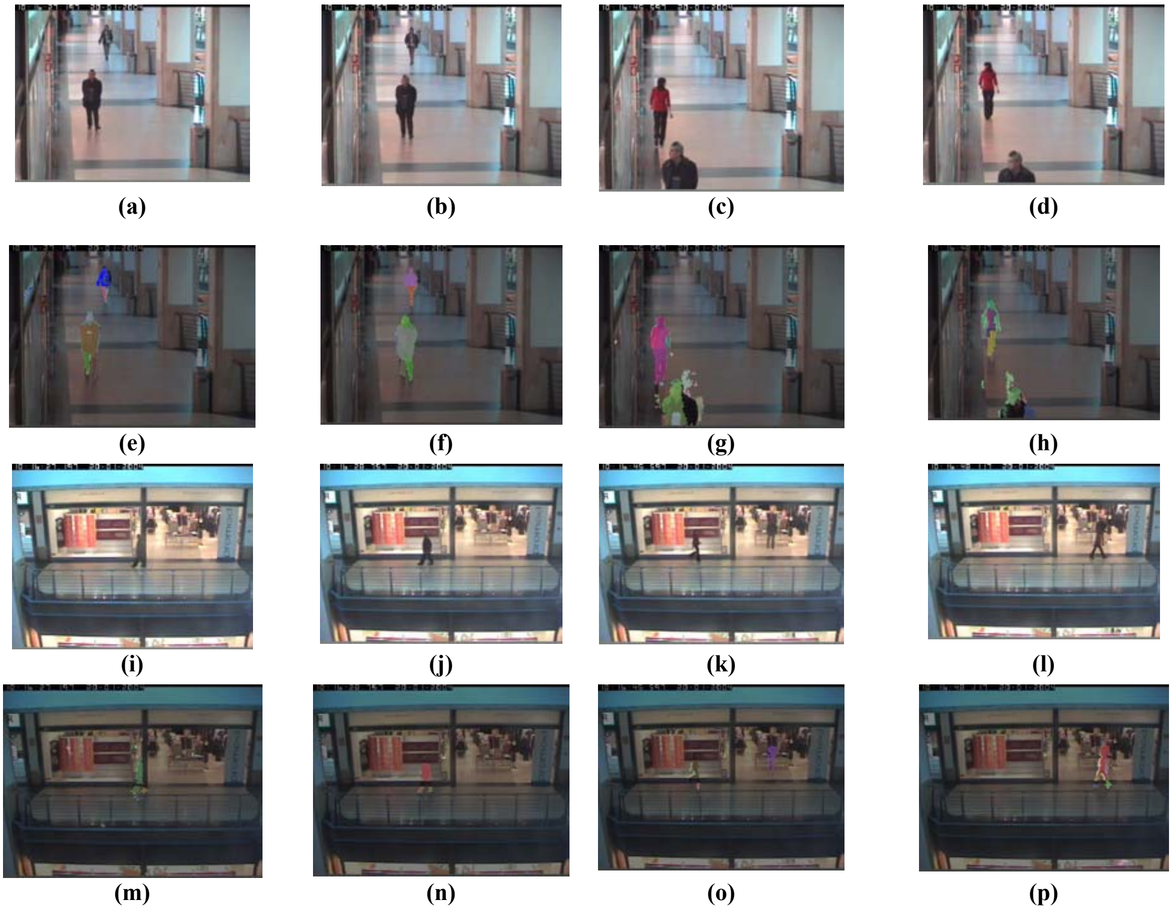


Fig. 10. Sample results obtained for "*OneStopNoEnter2*" data set of CAVIAR. (a-b) Original frames of corridor view. (e-h) Our segmentation results for (a-b). (i-l) Original frame of front view. (m-p) Segmentation results of (i-l). colors in segmented parts are chosen in random.

presented in Section 3.1 was obtained from these sequences as well.

Table 1 lists the complexity cost of our proposed method for moving object segmentation and Fuzzy color assignment processes. Our experimental result showed that our fast and accurate segmentation algorithm is superior to [14]. Obviously, the computational cost of the methods is highly dependent to the size of moving objects.

Table 1: Performance comparison of different moving object segmentation methods.

Video Type		Frame Size	Average elapsed time for each frame in seconds.	
			Method Proposed in [14]	Proposed Method
CAVIAR	TwoEnterShop1 corridor view	384×288	1.5167	0.0613
	TwoEnterShop1 front view	384×288	0.0634	0.0500
	OneStopNoEnter2 corridor view	384×288	1.3231	0.0827
	OneStopNoEnter2 front view	384×288	0.0534	0.0500
PETS 2001	TestSet1 Camera1	384×260	0.3458	0.0603
	TestSet1 Camera2	384×260	0.5765	0.0701

4.2. Tracking Results Using Fuzzy Region Adjacency Graph

To evaluate the performance of our tracker, plots shown in Fig. 11 are shown. The plot shows the tracking error of selected tracks from the first PETS2001 video sequence, and has been annotated to indicate the key events that have occurred during the period of tracking. In Figs. 11.c the blue and red colors show the error of occlusion solver using the Meanshift and proposed method, respectively. It can be observed that the largest errors occur when the object splits into several segments due to occlusion. Our proposed method has reduced the error by about 15.7%.

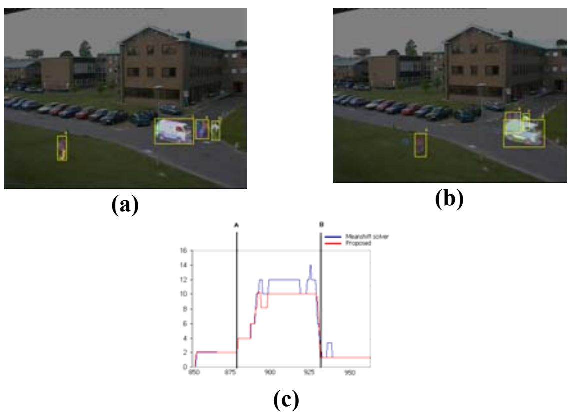


Fig. 11. Result of proposed tracker for Pets2001 dataset. a) Before occlusion. b) During occlusion. c) Error of occlusion solvers between frames 888 and 937: *blue*: Meanshift method, *red*: proposed method.

5. Conclusion

In this paper, we proposed a novel method to model moving objects in multicamera systems. Our proposed method constructs a region adjacency graph for each foreground object and solves the correspondence problem through a subgraph optimal isomorphism approach. A Fuzzified color histogram is then associated to each graph node as its attribute. Our proposed method is fast enough to be employed in real-time applications. Also, our proposed method is robust to face partial occlusion, color changes, and different object resolutions and orientations.

Acknowledgment

This work was in part supported by a grant from ITRC.

References

[1] T. Evgeniou, M. Pontil, and C. Papageorgiou, and T. Poggio, "Image representations for object detection using kernel classifiers," In Asian Conf. on computer vision. Pp.687-692, 2000.

[2] V. Kyrki, J.K Kamarainen, and H. Kälviäinen, "Simple Gabor feature space for invariant object recognition," Pattern Recognition Letters, Vol. 25, No. 3, pp. 311-318, 2004.

[3] M. Ulricha, C. Steger, and A. Baumgartner, "Real-time object recognition using a modified generalized Hough transform, " Pattern Recognition, Vol. 36,No. 11, pp.2557-2570, Nov. 2003.

[4] S.A Velastin, and P. Remagnino, Intelligent distributed video surveillance system, London: The institute of Electrical Engineers, 2005, pp. 164.

[5] T. Huang, and S.T Russell, Object Identification in a Bayesian Context, IJCAI, 1998, pp 1276-1283.

[6] V. Kettnaker, and R. Zabih, "Bayesian Multi-Camera Surveillance," in Conf. on computer vision and pattern recognition, pp.2253-2261, Jun. 1999.

[7] J. Orwell, P. Remagnino, and G.A. Jones, "Multiple Camera Color Tracking," in IEEE Intl. Workshop on visual surveillance, pp. 1355-1360, Oct, 2003.

[8] A. Elegmmal, and L.S. Davis, "Probabilistic framework for segmenting people under occlusion", in IEEE Intl. Conf. on Computer Vision,2001.

[9] O. Javed, Z.Rasheed, K. Shafique, and M. Shah, "Tracking Across Multiple Cameras with Disjoint Views", in 9th IEEE Int. Conf. on computer vision, pp. 952-957, Nice, France, 2003.

[10] A. Gilbert, and R. Bowden, "Incremental Modeling of the Posterior Distribution of Objects for Inter and Intra Camera Tracking," in British machine vision conference, Oxford, UK, 2005.

[11] S. Blackman, and R. Popoli, Design and analysis of modern tracking systems, Artech House, 1999.

[12] J. Kang, I. Cohen, and G. Mediono, "Continuous tracking within and across camera streams", in IEEE computer society Conf. on computer vision and pattern recognition, vol. 1, pp. 267-272, June. 2003.

[13] W. Zajdel, and B.J Krose, "A sequential Bayesian algorithm for surveillance with non-overlapping cameras", Int. Journal of pattern recognition and artificial intelligence. Vol. 9, No. 8, pp 977-996, Dec. 2005.

[14] H. Aghajan, and C. Wu, "Layered and collaborative gesture analysis in multi camera networks". In Int. Conf. on acoustics, speech and signal processing, Vol. 4, No. IV, pp 1377-1380, Honolulu, USA, Apr. 2007.

[15] M. Frank, H. Haag, H. Kollnig, and H.H. Nagel, ``Tracking of occluded vehicles in traffic scenes," in 4th European Conf. on computer vision, vol. 1065, pp. 485-494, Cambridge, UK, 1996.

[16] D. Koller, K. Daniilidis, and H.H Nagel, ``Model-based tracking in monocular image sequences of road traffic scenes," in Int. Journal of computer vision, Vol. 10, No. 3, pp. 257-281, 1993.

[17] A. Baumberg, and D. Hogg, "Learning flexible models from image sequences," in 3th European Conf. on computer vision, vol. 800, pp. 299-308, Stockholm, Sweden, 1994.

[18] D. Ramanan, D.A Forsyth, and A. Zisserman, "Tracking people by learning their appearance", in Trans. On pattern analysis and machine intelligence. Vol. 29, No. 1. Jan 2007.

[19] A. Jepson, D. Fleet, and T. El-Maraghi, "Robust online appearance models for visual tracking," IEEE trans on Pattern analysis and machine intelligence, Vol. 25, No. 10, pp. 1296-1311, Oct. 2003.

[20] A. Darabi, A.H. Khalili, and S. Kasaei, "Graph-based segmentation of moving object", in press on Int. GCC IEEE, Manama, Kingdom of Bahrain, Nov. 2007.

[21] P.F. Felzenswal, and D.T. Hutencheler, "Efficient Graph-Based Image Segmentation," in Int. Journal of computer vision, vol. 32, no. 14, 2004.

[22] B.K.P. Horn, and B.G. Schunck, "Determining Optical Flow," in Artificial Intelligence, Vol. 59, pp. 81-87, 1993.

[23] Y. El-Sonbaty, and M.A Ismail, "A new algorithm for subgraph optimal isomorphism," in Trans. On pattern recognition, Vol. 31, No. 2, pp. 205-218, 1998.

[24] Y. Bar-Shalom, and X. Li, Multitarget-multisensor tracking: principles and techniques, YBS publishing, 1995.

[25] EC Funded CAVIAR project/IST 2001 37540, found at URL: **http://homepages.inf.ed.ac.uk/rbf/CAVIAR/**.

[26] Found at URL http://peipa.essex.ac.uk/ipa/pix/pets/PETS2001/DATASET1/.