Supporting Information Reuse for Video Data Indexing by Using Color Information

Chengcui Zhang[1] Lin Luo[1] Shu-Ching Chen[1] Mei-Ling Shyu[2]
[1]Distributed Multimedia Information System Laboratory
School of Computer Science, Florida International University, Miami, FL 33199, USA
[2]Department of Electrical and Computer Engineering, University of Miami,
Coral Gables, FL 33124, USA

Abstract

In this paper, an object tracking framework assisted by image segmentation and color indexing techniques to support information reuse for video data is proposed. In order to obtain more accurate results of object tracking, a color indexing method called Color Label Criteria (CLC) is adopted to relax some of the restrictions and assumptions in object tracking in our previous work with low additional cost. Our experiment results on a real video sequence show that the CLC color indexing method is very helpful to achieve more robust object tracking.

1. INTRODUCTION

Nowadays, networked computers have access to huge amounts of potentially useful new information every day, which as a result intensifies the challenges of extracting data, and reusing it in varied applications and settings. The diversity of multimedia information, both in its structure and content, is the main barriers to its reuse and maintenance. All large-scale multimedia information systems require their information to be pre-formatted or pre-indexed before any kind of further processing on it can be executed. Extracting information and indexing the information into the format suitable for processing is often the most important aspect of multimedia information reuse and integration.

Recently, digital video has been widely used in many multimedia applications such as education, training, video on demand, and video conferencing. How to structure and index video data to allow users to quickly and easily retrieve interesting materials becomes an important issue in multimedia information reuse. In this paper, in order to better index the video data for efficient reuse such as multimedia database queries, an object tracking framework assisted by image segmentation and color indexing is proposed to extract and index the spatio-temporal relationships of objects in the video sequences. To answer multimedia database queries related to the temporal or relative spatial positions of semantic objects, it is necessary to have object-based representation of video data. For this purpose, the Simultaneous Partition and Class Parameter Estimation (SPCPE) algorithm [1, 2] is used to capture the temporal and spatial relations of the semantic objects (such as the players in a soccer game video). Instead of using the low-level features such as color and texture [3, 4, 5], the SPCPE method is one kind of statistical method attempting to achieve global minimization of locally defined cost functions. Based on the segments/objects obtained by the segmentation process, object tracking can be done in a relatively simple way.

Lots of research work has been proposed towards object tracking [8, 9, 10]. An adaptive Gaussian mixture model is used to track the object in [8]; while an affined structure for point correspondences within the objects is employed in [9]. In [10], object tracking is performed based on optical flow and depth. What was missing in their work is that the initial object information, such as the spatial location and region, is not readily available so that they have to manually select the region of target object (e.g., the rectangular region) they want to track, which is the impediment to enable the automatic information extraction and indexing. With good segmentation results, rich knowledge such as spatial locations and areas of objects can be obtained for future tracking.

The proposed framework tries to achieve automatic object tracking in video sequences. Moreover, in order to obtain more accurate results of object tracking, a color indexing method is adopted to relax some of the restrictions and assumptions in object tracking in our previous work [1] with low additional cost. In this paper, a color indexing method called Color Label Criteria (CLC) is applied on each identified object. The experimental results based on a real soccer game video are presented to demonstrate the effectiveness of the proposed framework.

In what follows we describe the proposed framework to object tracking. In Section 2, the SPCPE algorithm and basic workflow of object tracking are introduced. The enhanced object tracking assisted by color indexing is discussed in Section 3. Experimental results are shown in Section 4. Section 5 gives the conclusion and future work.

2. SPCPE ALGORITHM AND OBJECT TRACKING

2.1 Overview of the SPCPE Algorithm

The SPCPE (Simultaneous Partition and Class Parameter Estimation) algorithm is an unsupervised video segmentation method to partition video frames. A given class description determines a partition. Similarly, a given partition gives rise to a class description, so the partition and the class parameter have to be estimated simultaneously. In practice, the class descriptions and their parameters are not readily available. An additional difficulty arises when images have to be partitioned automatically without the intervention of the user. In the SPCPE algorithm, the partition and the class parameters are treated as random variables. The method for partitioning a video frame starts with an arbitrary partition and employs an iterative algorithm to estimate the partition and the class parameters jointly [2, 7]. Since the successive frames in a video do not differ much, the partitions of adjacent frames do not differ significantly. Each frame is partitioned by using the partition of the previous frame as an initial condition so the number of iterations in processing can be greatly reduced. A randomly generated initial partition is used for the first frame since no previous frame is available.

The mathematical description of a class specifies the pixel values as functions of the spatial coordinates of the pixel. The parameters of each class can be computed directly by using the least squares technique. Suppose we have two classes. Let the partition variable be $c = \{c_1, c_2\}$ and the classes be parameterized by $\theta = \{\theta_1, \theta_2\}$. Also, suppose all the pixel values y_{ij} (in the image data Y) belonging to class k $(k=1,2)$ are put into a vector $\boldsymbol{Y_k}$. Each row of the matrix Φ is given by $(1, i, j, ij)$ and a_k is the vector of parameters $(a_{k0}, \ldots, a_{k3})^T$.

$$y_{ij} = a_{k0} + a_{k1}i + a_{k2}j + a_{k3}ij, \ \forall(i, j)\ y_{ij} \in \boldsymbol{c_k}$$

$$\boldsymbol{Y_k} = \Phi\, a_k$$

$$\hat{a}_k = (\Phi^T\Phi)^{-1}\Phi^T \boldsymbol{Y_k}$$

We estimate the best partition as that which maximizes the a posteriori probability (MAP) of the partition variable given the image data Y. Now, the MAP estimates of $c = \{c_1, c_2\}$ and $\theta = \{\theta_1, \theta_2\}$ are given by

$$\begin{aligned}(\hat{c},\hat{\theta}) &= Arg \max_{(c,\theta)} P(c,\theta \mid Y) \\ &= Arg \max_{(c,\theta)} P\ (Y \mid c,\theta) P(c,\theta)\end{aligned}$$

Let $J(c,\theta)$ be the functional to be minimized. With appropriate assumptions, this joint estimation can be simplified to the following form:

$$(\hat{c},\hat{\theta}) = Arg \min_{(c,\theta)} J(c_1, c_2, \theta_1, \theta_2)$$

$$J(c_1,c_2,\theta_1,\theta_2) = \sum_{y_{ij}\in c_1} -\ln p_1(y_{ij};\theta_1) + \sum_{y_{ij}\in c_2} -\ln p_2(y_{ij};\theta_2)$$

It may appear as though the simultaneous minimization of J on c and θ is hard, but it is not. We will show how the minimization of J can be carried out alternately on c and θ in an iterative manner. Let $\hat{\theta}(c)$ represent the least squares estimates of the class parameters for a given partition c**.** The final expression for $J\left(c,\hat{\theta}(c)\right)$ can be derived easily and is given by

$$J\left(c,\hat{\theta}(c)\right) = Arg \min_{(c_1,c_2)} \left\{ \frac{N_1}{2} \ln \hat{\rho}_1 + \frac{N_2}{2} \ln \hat{\rho}_2 \right\}$$

where $\hat{\rho}_1$ and $\hat{\rho}_2$ are the estimated model error variances of the two classes.

The algorithm starts with an arbitrary partition of the data and computes the corresponding class parameters. With these class parameters and the data, a new partition is estimated. Both the partition and the class parameters are iteratively refined until there is no further change in them.

After the segmentation, the minimal bounding rectangle (MBR) concept in R-tree [6] is adopted so that each semantic object is bounded by a rectangle. Moreover, the centroid point of each semantic object is mapped to a point object for spatial reasoning. A segmentation example will be given in Section 3.2.

2.2 Object Tracking

The object-tracking method is used to track the objects (segments) within the successive video frames so that the related objects in successive frames can be tracked and grouped together to form the trail of that object. Intuitively, two segments in adjacent frames are considered as "related" based on their spatial closeness. The spatial closeness measure consists of two parts: 1) the distance between the centroids of two segments; 2) the areas of their bounding boxes.

Euclidean distance function is used to measure the distance between two centroids. Let ctr_p and ctr_c denote the centroids of two segments p and c in previous and current frame respectively. The distance is defined as:

$$dist(ctr_p - ctr_c) = \| ctr_p - ctr_c \|_2 \leq \delta,$$

where the threshold δ is 10 pixels in our experiments.

Also, in our previous work [1], we assume that the size change of two segments in successive frames should not be large if they represent the same object. Let BB_p and BB_c be the bounding boxes of segments p and c. A bounding box BB is a rectangle and represented as:

$$BB = (X_{tl}, Y_{tl}, X_{br}, Y_{br}),$$

where (X_{tl}, Y_{tl}) and (X_{br}, Y_{br}) are the coordinates of the upper left vertex and the lower right vertex of that bounding box, respectively.

The area of bounding box BB (i.e., the number of pixels included in that bounding box) is defined as the following equation.

$$area(BB) = (X_{br} - X_{tl}) \times (Y_{br} - Y_{tl}).$$

The size restriction for the area is

$$\left| \frac{area(BB_p) - area(BB_c)}{\max(area(BB_p), area(BB_c))} \right| \leq \beta,$$

where $\beta = 0.32$ in our experiments.

If segments p and c satisfy both of the distance and area restrictions, they are identified as related segments, which means they are the representations of the same object in different frames. Thresholds δ and β are selected based on results of our experiments. Considering the sampling rate of the video sequence and the object's moving speeds, the higher the sampling rate, the smaller the values of δ and β; while the faster the objects move, the bigger the values of δ and β. As described above, this object tracking schema is simple and effective in many cases since the spatial locations and areas of objects have already been obtained through the process of segmentation. However, since the threshold values of δ and β are fixed throughout the whole tracking process, there may exist some situations that are difficult for this basic object tracking method to work well. For example, in a soccer game video, if one player object suddenly fell into the ground, the position of its centroid and/or its area may change a lot in consecutive frames, which may result in identifying them as non-related segments that actually correspond to the same object. There are two possible ways to overcome this problem. First, the self-adaptive threshold values can be used. Second, some low-level features of an object can be used to help object tracking. In this paper, the latter approach is adopted by using the color indexing method to help identify the corresponding segments in consecutive frames. The details will be discussed in the next section.

3. COLOR INDEXING

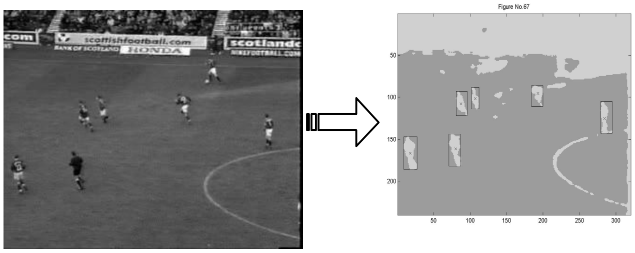
3.1 Color Information

Color is an important feature of the images. For the purposes of storage and display, color images are usually represented and handled in the RGB format. An RGB image, sometimes referred to as a "true color" image, can be represented as a set of 3 *m*-by-*n* matrices that define the red, green, and blue color components for each individual pixel, where *m* and *n* are the width and height of an image in pixel units.

3.2 Color Feature Extraction

Before introducing the color information into the proposed object-tracking algorithm, the color feature in the video frames needs to be extracted. In our experiments, the RGB color space is used as the experimental color space.

In our previous work [1], the RGB channels of the color video frames are averaged and the average value is the input for the segmentation process. Then the SPCPE algorithm is applied to get a sequence of segmented frames that are called the "*segmentation maps*". Each map contains a group of identified objects (segments). Then these objects (segments) are represented with their bounding boxes and centroids. An example of the segmentation process is shown in Figure 1. The left one is the original video frame, and the right one is the segmentation map for that frame together with the bounding boxes and centroids for the semantic objects (such as the soccer players) in that frame. As shown in Figure 1(b), the light gray areas in the segmentation map represent the objects, while the dark areas represent the ground.



(a) Original video frame (b) Segmentation map

Figure 1: Illustration of Segmentation Process

As mentioned earlier, there are size restriction of the bounding boxes and the Euclidean distance restriction of their centroids in our previous object-tracking method. In this paper, we try to relax these two restrictions by taking the advantage of color information of each object.

Color histograms are a simple non-parametric method for modeling. For each object (segment) identified by the SPCPE algorithm, the three-color components (r, g, b) of each pixel within the shape of that object are extracted, and then the 52-bins histogram of three different color channels is drawn. The purpose is to extract several typical colors that can be used to represent that object. For each object *i*, the following two color labels are assigned.

$$CL_{i1} = (r_{i1}, g_{i1}, b_{i1}) \text{ and } CL_{i2} = (r_{i2}, g_{i2}, b_{i2}),$$

where r_{i1}, g_{i1}, b_{i1} (r_{i2}, g_{i2}, b_{i2}) are the indexes of the bins that contain the largest (the second largest) numbers of pixels in color channels R, G, B respectively. In other words, CL_{i1} and CL_{i2} contain the indexes, from the three individual histograms, of bins which contain the largest and the second largest numbers of pixels. Segments intend to keep the similar color labels if they are the appearances of the same object in the successive frames. Based on the result of our experiments, two criteria related to color information called Color Label Criteria (CLC) are used. If the color labels of two segments p and c in the previous and current frames satisfy **CLC1**, then these two segments are considered related; or if **CLC1** is not satisfied but **CLC2** can be satisfied, these segments are also considered related.

(**CLC1**)**:** For the color labels CL_{p1} and CL_{c1} that record the indexes of the bins containing the largest numbers of pixels in the *R*, *G*, *B* channels, the differences of all three bin indexes should within the range [-2, 2]. That is,

$$|r_{p1} - r_{c1}| \leq 2 \ and \ |g_{p1} - g_{c1}| \leq 2 \ and \ |b_{p1} - b_{c1}| \leq 2 .$$

(**CLC2**)**:** For the color labels CL_{p1} and CL_{c2}, and CL_{p2} and CL_{c1}, each pair of the color labels should satisfy **CLC1**. In other words,

$$|r_{p1} - r_{c2}| \leq 2 \ and |g_{p1} - g_{c2}| \leq 2 \ and |b_{p1} - b_{c2}| \leq 2 \ ;$$

and

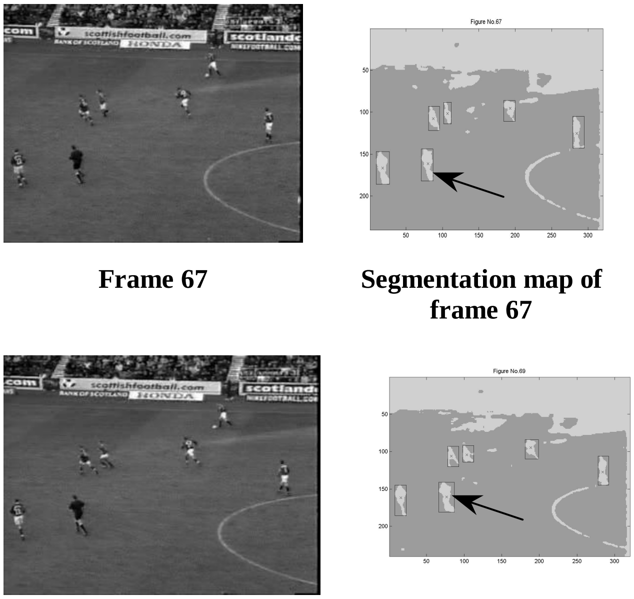
$$|r_{p2} - r_{c1}| \leq 2 \ and |g_{p2} - g_{c1}| \leq 2 \ and |b_{p2} - b_{c1}| \leq 2 .$$

4. EXPERIMENTAL RESULTS

A real soccer game video is used in the experiments to demonstrate the effectiveness of the proposed framework. In this experimental soccer video sequence, the semantic objects (segments) of interest are the individual soccer players and the referee.

By introducing color features into our object-tracking method, we can identify the related objects that are unable to be identified by the original object tracking method proposed in [1]. As shown in Figure 2, according to the original color images of frames 67 and 68, the two bounding boxes pointed by the arrows in the segmentation maps should be the same person (the referee) in the adjacent frames. However, the area of bounding box for the referee in frame 67 is 624 (in pixel units) and the area of the bounding box in frame 68 is 902 (in pixel units) so that the ratio of the area change is:

$$\left| 624 - 902 \right| \ / \ 624 \approx 0.44 .$$



Frame 67

Segmentation map of frame 67

Frame 68

Segmentation map of frame 68

Figure 2: An example of using color indexing to improve object tracking

Such area change exceeds the threshold value 0.32 for the size. Therefore, these two objects cannot be identified as “related” in our original object tracking method. However, when taking into account of color information, these two objects are successfully identified as related in consecutive frames. Their CL_1’s are exactly the same so that **CLC1** is satisfied. The color labels assigned to those two objects pointed by the arrows in

Frames 67 and 68 (as shown in Figure 2) are listed in Table 1.

Table 1: The color labels for the object (pointed by the arrow) to be tracked in Figure 2

	$CL_1(r, g, b)$	$CL_2(r, g, b)$
Frame 67	(4, 4, 1)	(14, 5, 2)
Frame 68	(4, 4, 1)	(3, 15, 11)

5. CONCLUSIONS AND FUTURE WORK

In this paper, a color-assisted object tracking framework to index the objects in video data for multimedia database queries to allow multimedia information reuse is presented. In the proposed framework, an unsupervised segmentation method is applied first to extract the spatial-temporal relations of the objects in video sequences. Then, the Color Label Criteria (CLC) color indexing method is proposed to improve the robustness of object tracking. It should be pointed out that the processing of CLC can be integrated into the segmentation process. Hence, the additional cost introduced by that is very small. The experimental results on a real soccer game video sequence are presented to demonstrate the effectiveness of the proposed framework. Currently, the computation for our segmentation and object tracking algorithm is efficient, so it is possible to process the segmentation and object tracking in real time.

Our future work will focus on multimedia database queries that can be answered based on the current framework. In the framework, video data can be represented in an object-based representation. Moreover, an efficient data structure for storing and retrieving of the indexed information will be further explored.

6. ACKNOWLEDGEMENT

For Shu-Ching Chen, this research was supported by in part by NSF CDA-9711582.

7. REFERENCES

[1] S.-C. Chen, M.-L. Shyu, C. Zhang, and R. L. Kashyap, “Object Tracking and Augmented Transition Network for Video Indexing and Modeling,” *12th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2000)*, pp. 428-435, November 13-15, 2000, Vancouver, British Columbia, Canada.

[2] S. Sista and R. L. Kashyap, “Unsupervised video segmentation and object tracking,” in *IEEE Int’l Conf. on Image Processing*, 1999.

[3] W. Skarbek and A. Koschan, “Color Image Segmentation –A Survery,” Technical report 94-32, Computer Science Department, TU Berlin, 1994.

[4] Y. Raja, S. McKenna, and S. Gong, “Segmentation and tracking using colour mixture models,” In *Asian Conference on Computer Vision*, 1998.

[5] D. Comaniciu and P. Meer, “Robust analysis of feature space: color image segmentation,” *Proc. Of IEEE Conf. On Computer Vision and Pattern Recognition*, pp 750-755, 1997.

[6] Guttman, “R-tree: A Dynamic Index Structure for Spatial Search,” in *Proc. ACM SIGMOD*, pp. 47-57, June 1984.

[7] S. Sista and R. L. Kashyap, “Bayesian Estimation for Multiscale Image Segmentation,” in *IEEE Int’l Conf. on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, March 1999.

[8] S. J. McKenna, Y. Raja and S. Gong, “Object Tracking Using Adaptive Colour Mixture Models,” *Lecture Notes in Computer Science*, 1(1351): 615-622, 1998.

[9] G. S. Manku, P. Jain, A. Aggarwal, L. Kumar and S. Banerjee, “Object Tracking using Affine Structure for Point Correspondences,” *Proc. IEEE Conf. for Computer Vision and Pattern Recognition*, pp. 704-709, June 17-19, 1997, Puerto Rico.

[10] R. Okada, Y, Shirai and J. Miura, “Object Tracking Based on Optical Flow and Depth,” *Proc. of IEEE/SICE/RSJ Int. Conf. on MFI*, pp.565-571, 1996.