Spatio-temporal Relationships and Video Object Extraction

Yining Deng and B. S. Manjunath
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106-9560
{deng, manj}@iplab.ece.ucsb.edu

Abstract

An object-based representation for video data can facilitate video search and content analysis. Detecting physical meaningful video object is a challenging open issue, and requires intelligent spatio-temporal segmentation and tracking. Normally, this is done through spatio-temporal segmentation and region tracking. In this work, some of the practical issues of segmentation and tracking problems are addressed. Due to the limitation of using low-level visual features in the segmentation, the tracked regions are more likely to be fragmented parts of some meaningful objects. However, if a collection of video shots that contain a particular object of interest are given, spatio-temporal correlations would exist between the neighboring regions of the object. A method of mining association rules is used to discover these patterns and thus to find possible objects in the scene. Initial experimental results of this approach are shown.

1. Introduction

The rapid developments in multimedia and internet applications allow us to conveniently access large amounts of video data in the new digital format. One can now download news and movie clips directly from the internet, and watch the movies on digital versatile discs (DVD) which provide much clearer pictures than the traditional video tapes. While there is significant progress in compression to represent the video in compact form for storage and transmission, functionalities that come with the digital video have not been fully exploited. For example, finding desired information in a video clip or in a video database is still a difficult and laborious task. There is a growing need for new representations of video that allow not only compact storage of data but also content-based functionalities such as search and manipulation of objects, semantic description of scenes, detection of unusual events, or recognition of objects.

A video clip, whether in raw format or in one of the current compression standards, is a binary stream not well organized in terms of its content. As a first step toward better organization of data, the video clip is parsed in temporal domain into short video shots. Within each shot the visual appearance of the scene is consistent and the partitioning points are often camera breaks or editing cuts. A video shot can be considered as a basic unit of video data. Global image features such as color, texture, and motion can be extracted and used for search and retrieval of similar video shots. Most of the video content-based retrieval systems have been using this kind of approach [1] [6] [9] [10] [11] [13] [15].

The above approach, however, provides very limited video information and can not answer simple questions such as "what kind of *things* are in the video". In order to further exploit the video content, a video shot needs to be decomposed into objects so that search, retrieval, and content manipulation based on object characteristics, activities, and relationships is possible. Although this could be done manually such as in the work of [2], the lengthy and tedious laboring makes it impossible in many applications. In our previous work [7], an object-based representation is proposed for search and retrieval of video objects. Practical issues of using automatic spatio-temporal segmentation and region tracking to extract video objects are addressed. There are a few systems reported recently that have also proposed similar approaches [14] [16]. However, the disadvantage problem arises due to the limitation of using low-level visual features in the segmentation. The extracted components are more likely to be segments of meaningful objects and their direct usefulness is limited.

It is possible to extract the meaningful objects given the knowledge of a constraint environment. For example, in [12], knowledge of a player's size and the color of a soccer field is assumed to identify the players and the ball in a soccer video database. In [8], moving objects are extracted based on the assumption that the background is static and the objects are relatively small compared to the background. These facts allow the background registration

This work was supported in part by an award from Samsung Electronics.

0-7803-5148-7/98/$10.00©1998 IEEE

when the camera is moving and also imply motion coherence for each object. Although a similar motion coherence constraint can be used on those fragmented regions, it is in general not true that parts of a non-rigid object will undergo the same motion when the object appears large in the scene. Moreover, if the objects are in a changing background, the task of identify objects in the video scene is very difficult.

In this work, we approach the problem from a database perspective. If a collection of video shots that contain a particular object of interest are given, spatio-temporal correlations would exist between the neighboring regions of the object. A method of mining association rules [4] is proposed to discover these patterns and thus find possible objects in the scene. This kind of approach has also been used in image retrieval [3]. It is to be noted that although the approach is to explore the spatio-temporal relationships in the video, our objective is different from the traditional sense where the objects are known and relationships among the objects are being sought, such as in [2], [5], and [12].

In the next section a summary of our previous work on spatio-temporal segmentation and region tracking is given. Section 3 describes the method of mining association rules and the application to the object extraction problem. Section 4 provides results and discussions.

2. Segmentation and Tracking

2.1 Spatio-temporal Segmentation

General Scheme: Reliable spatio-temporal segmentation of objects is a difficult problem. One of the popular approaches is to use optical flow method, which has the following drawbacks:

- It does not work well with large motion.
- Regions of coherent motion may contain multiple objects and need further segmentation for object extraction, for example, a person riding a horse.

To overcome these drawbacks, it is important to incorporate spatial information into motion segmentation. One feasible approach is to spatially segment the first frame to obtain initial segmentation results, and then motion segment subsequent frames using affine region matching. There are several advantages of doing this:

- Multiple objects with the same motion are separated by spatial segmentation. These objects can still be merged together for analysis if necessary after motion estimation.
- Affine region matching is a more reliable way of estimating motion than optical flow methods and there are some fast numerical methods proposed to estimate the affine motion parameters.

Problems remain for this approach to work in practice. The new objects entering the scene and the propagation error due to affine region matching must be handled. In the following, we propose to use a group processing scheme similar to the one employed in MPEG-2 to refresh the spatial segmentation and ensure the robustness of the algorithm.

Video data is processed in consecutive groups of frames. These groups are non-overlapping and independent of each other. The best value for the number of frames in a group depends on object motion activities in the video data. There is one I-frame in each group, which is spatially segmented first. Starting from the I-frame, the rest of the frames in the group are segmented consecutively by affine matching the segmented regions to their next frame. These motion segmented frames are called P-frames.

Motion Segmentation: After spatial segmentation, a 6-parameter 2D affine transformation is assumed for each region in the frame and is estimated by finding the best match in the next frame. Consequently, segmentation results for the next frame is obtained. The following functional f is to be minimized for each region R,

$$f(a) = \sum_{(x, y) \in R} g(I_1(x', y') - I_2(x, y)) \qquad (1)$$

where a is a six-parameter affine motion vector, g is a robust error norm to reject outliers, I_1 and I_2 are the current frame and the next frame respectively, x and y are pixel locations, $x' = x + dx$ and $y' = y + dy$, dx and dy are displacement vectors.

The minimization of f can be carried out iteratively by gradient based methods. This requires the cost function to be convex in affine space to guarantee convergence to the global minimum. We assume this to be true in the vicinity of the actual affine parameter values. Thus a good initialization is needed before starting the iterations. In the case that affine parameters of the previous frame are known, we can make a first-order assumption that the region is going to keep the same motion and use the affine parameters of the previous frame as the initial values for the current frame. For the first frame to be segmented by motion prediction in each group, whose previous frame is an I-frame, a hierarchical search is performed to obtain the best initial affine parameters.

2.2 Region Tracking

After segmentation, regions in the video frames are tracked to locate object movements. The processes of tracking and local feature extraction are closely related. One advantage of using the proposed segmentation scheme is that the problem of region tracking is simplified. There

are two types of region tracking, intra-group region tracking and inter-group region tracking. Within each group, region tracking is already done during the segmentation. This accounts for most of the frames in the video and reduces the complexity of tracking. The remaining problem is to track regions across two adjacent groups. Because the regions could differ significantly in two groups due to large object motion, it is impossible to match all the regions between the two groups. Therefore, we do not attempt to match every region in the current group to the regions in the next group.

The I-frame is the representing frame of each group. The tracking is performed between consecutive I-frames by comparing similarity between regions in the two frames using extracted local features, including color, texture, size, and location. Motion features are used to predict region locations. For a region R in the current I-frame, the steps for finding the matching region in the next I-frame are as follows:

1. For every region in the next I-frame, the size and texture differences with R are calculated. If they are less than the preset thresholds, the region is considered as a candidate match.
2. Affine motion is estimated to predict the approximate location of R in the next I-frame. The distance between the location of a candidate region and the predicted location of R is calculated and is denoted by d_L.
3. The color feature distance between a candidate region and R is weighted by $1 + W_L d_L$, where W_L is a constant. That is, the further away a candidate region, the less likely that it will be the true match. The particular weighting by location difference is needed because there might be some objects nearby with similar color, texture, and size.
4. The candidate regions are ranked in terms of d_C. The region having the minimum value is determined to be the match for R.

3. Spatio-temporal Relationships

The problem of the above approach is that segmentation based on low-level visual features usually results in many fragmented regions. As can be seen from the examples in Fig. 1, one cause of fragmentation is that an object contains multiple parts of different colors, for example, Miss America's body in (a) and the table in (b). Another cause is due to the shades created by the lighting conditions, for example, the hair in (a) and the arm in (b). The fragmentation not only causes difficulty in tracking, but also makes the tracking results less useful.

In this work, we approach this problem from a database perspective. Suppose the database consists of a collection of video shots that contain a particular object of interest. Correlations would exist between the neighboring segmented regions such that certain regions tend to appear together at the same time. If this kind of pattern is found, it can be inferred that these regions could belong to the same object. Several conditions are required for this idea to work in practice:

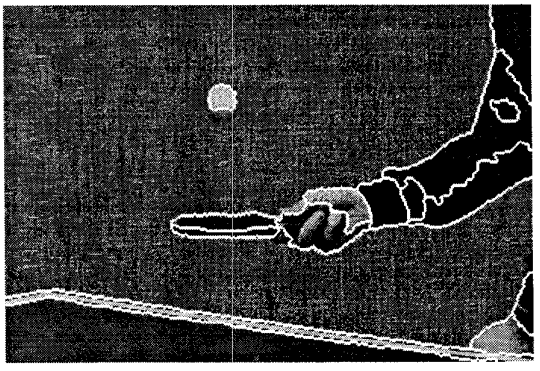
- The visual appearance of the object, such as the color information, remains the same in the database.
- The visual appearance of the background, on the other hand, varies from shot to shot.
- The object is distinguishable from the background, i.e., the background does not have the same color or texture as the object.

To find the spatio-temporal correlations of the neighboring regions, the method of mining association rules is used. In the following, the concept of association rules and how to discover the rules will be explained.

3.1 Mining Association Rules

Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of items. Let D be a

(a)



(b)

Fig 1. Segmentation results of, (a) "Miss America" sequence (frame 130), (b) "table tennis" sequence (frame 0).

set of events, where each event T is a set of items such that $T \subseteq D$. Let X be a set of items. An event T is said to contain X if and only if $X \subseteq T$. An association rule is an implication of the form $X \Rightarrow Y$, where $X \subseteq I$, $Y \subseteq I$, and $X \cap Y = \phi$. The rule $X \Rightarrow Y$ holds in the event set D with *confidence* c if $c\%$ of events in D that contain X also contain Y. The rule $X \Rightarrow Y$ has *support* s in the event set D if $s\%$ of events in D contain $X \cup Y$. *Confidence* denotes the strength of implication and *support* indicates the frequencies of the occurring patterns in the rule. *Confidence* c is in fact a measure of conditional probability $P(Y|X)$. Rules with high *confidence* and strong *support* are referred as strong rules.

Although it is quite a simple concept, discovering strong rules in large databases is not an easy task. The problem is usually decomposed into two steps:

1. Discover the large itemsets, i.e., the itemsets that have strong *support* in the database.
2. Use the large itemsets to generate the association rules.

Clearly, Step 1 is much more critical and algorithms have been proposed [4] for efficient counting of large itemsets. A common approach is to start with large 1-itemsets, i.e., the itemsets that contain only 1 item, and then increase the number of items to k to discover large k-itemsets.

3.2 Application to Object Extraction

In order to discover the spatio-temporal correlations of the neighboring regions, the method of mining association rules is first applied to the segmented regions within each video shot. The discovered rules are then verified over a collection of shots in the database.

Shot Level: The procedures for rule discovery at shot level are summarized as follows:

1. *obtaining the database items* - The segmented regions can not be used directly yet because they are not in the sense of database items mentioned earlier. These regions need to be classified and each class of regions is considered as an item. The k-means algorithm is used to cluster the regions based on their low-level features. The average number of regions per frame in the video shot is chosen as the initial number of classes. An agglomerative clustering algorithm is used later to further merge close clusters.
2. *finding the large 1-itemsets* - To remove outliers due to segmentation failure or over-segmentation, if the distance between the region feature and its cluster centroid is too large, the region is removed and the centroid is recalculated to obtain the feature for that class. The class size is defined to be the number of the regions in the class. This number is counted for each class. Simultaneous occurrences of a class in a video frame are counted only once. If the number exceeds 70% (*support*) of the number of frames in the video shot, the class is considered as a large 1-itemset, otherwise the class is discarded.
3. *finding the large 2-itemsets* - The neighboring relationships of the classes are determined here. The number of occurrences of two classes being neighboring to each other is counted. Again simultaneous occurrences in a video frame are counted once. If the number exceeds 70% (*support*) of the number of the frames in the video shot, the two classes are considered to be neighbors to each other and a large 2-itemset is found.
4. *discovering the strong rules* - For each class, the size of each neighboring class is checked to see if it exceeds 80% of the size of that class. If true, an association rule is stored. Else, that neighbor is removed from the list of the neighbors.

Database Level: It is quite likely that within a given video shot the background may remain constant. In such cases, the discovered association rules can imply false correlations between the objects and the background. This can be verified and removed by analyzing over a collection of video shots where the background is different:

1. *finding the classes in interest* - For each class in a video shot, if similar classes can be found in other shots, the class is considered as a possible part of an object in interest. The similarity is measured based on class features calculated at shot level.
2. *verifying the association rules* - For each class of interest, its neighboring classes are compared with the ones of similar classes in other video shots. If the number of classes containing a similar neighbor exceeds 60% of the number of video shots, the association rule is considered to be significant. Regions that possibly belong to a same object are then identified.

4. Results and Discussions

Initial experiments have been carried out on a collection of 9 cartoon video shots, which all contain a specific animal. The video is at a rate of 30 frame/sec and the average number of frames in the video shots is about 114. There are a total of 28821 segmented regions and 124 classes with strong *support* in the entire collection. Region color features are used in the mining association rule discovery. There are 11 rules discovered in 6 of these 9 shots. Table 1 summarizes the results, where in the third case a background region with similar colors is merged into the same class of an object part such that the class contains both the object part and the background region.

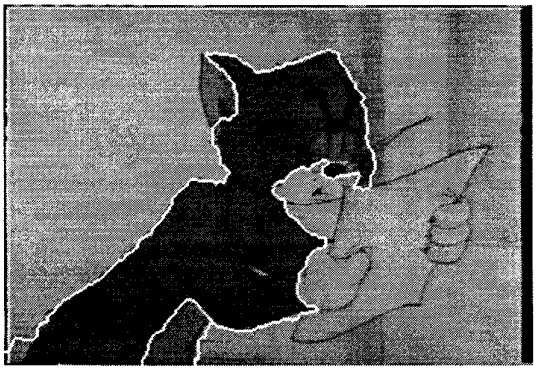
It can be seen from the results that the discovered rules capture a certain degree of spatio-temporal correlations between the object regions. Fig. 2 shows two examples of

Table 1: Discovered Strong Associated Rules

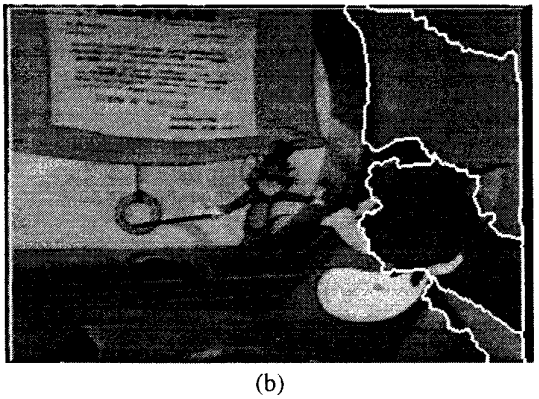
actual relationships between the classes	number of rules
both are parts of the object (correct rules)	4
a part of the object - a background region	2
a part of the object - object and background	2
both are background regions	3

discovered associated regions. In (a), a correct associated rule that identifies the main and the front bodies of an animal is found. In (b), the same rule is discovered, but the front body is mixed up with a background region as the same class.

The false alarms and the misses are mainly due to the fact that required conditions mentioned before are not met. The object appearance actually varies from shot to shot under different lighting conditions. Some of the background colors appear almost in all the video shots. Also there are cases where the background regions are very similar to the object parts. Note that these situations happen in practice and will cause problems in object extraction even when the prior knowledge of the object, such as the color information, is known. Adding the region motion information to cross check the results could possibly eliminate some of the false alarms.



(a)



(b)

Fig 2. Pairs of the associated regions. The two object parts discovered are the main and the front bodies of an animal.

5. References

[1] E. Ardizzone, M.L. Cascia, "Multifeature image and video content-based storage and retrieval", *Proc. of SPIE*, vol. 2916, p. 265-276, 1996.

[2] A.B. Bimbo, E. Vicario, and D. Zingoni, "Symbolic description and visual querying of image sequences using spatio-temporal logic", *IEEE Trans. on Knowledge and Data Engineering*, vol. 7, no. 4, p. 609-22, Aug., 1995.

[3] M. Bouet and C. Djeraba, "Powerful image organization in visual retrieval systems", *Proc. ACM Multimedia*, 1998.

[4] M.-S. Chen, J. Han, and P.S. Yu, "Data mining: an overview from a database perspective", *IEEE Trans. on Knowledge and Data Engineering*, vol. 8, no.6, p. 866-83, Dec., 1996.

[5] Y. F. Day, et. al., "Spatio-temporal modeling of video data for on-line object-oriented query processing", *Proc. of IEEE Intl. Conf. Multimedia Computing and Systems*, p. 98-105, 1995.

[6] Y. Deng and B.S. Manjunath, "Content-based Search of Video Using Color, Texture and Motion", *Proc. of ICIP*, vol. 2, p. 534-37, 1997.

[7] Y. Deng and B.S. Manjunath, "NeTra-V: toward an object-based video representation", *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, no. 5, p. 616-27, Sep., 1998.

[8] G. Halevi and D. Weinshall, "Motion of disturbances: detection and tracking of multi-body non-rigid motion", *Proc. of CVPR*, p. 897-902, 1997.

[9] A. Hampapur, et. al., "Virage video engine", *Proc. of SPIE*, vol. 3022, p. 188-200, 1997.

[10] G. Iyengar and A.B.Lippman, "Videobook: an experiment in characterization of Video", *Proc. of ICIP*, vol.3, p. 855-58, 1996.

[11] V. Kobla, D. Doermann, and K. Lin, "Archiving, indexing, and retrieval of video in the compressed domain", *Proc. of SPIE*, vol. 2916, p. 78-89, 1996.

[12] A. Yoshitaka, et.al., "Content-based retrieval of video data based on spatiotemporal correlation of objects", *Proc. of IEEE Intl. Conf. Multimedia Computing and Systems*, p. 208-13, 1998.

[13] H.J. Zhang, J. Wu, D. Zhong and S.W. Smolliar, "An integrated system for content-based video retrieval and browsing", *Pattern Recognition*, vol. 30, no. 4, p. 643-58, 1997.

[14] H.J. Zhang, J. Y.A. Wang and Y. Altunbasak, "Content-based video retrieval and compression: a unified solution", *Proc. of ICIP*, vol. 1, p. 13-16, 1997.

[15] D. Zhong, H.J. Zhang and S-F. Chang, "Clustering methods for video browsing and annotation", *Proc. SPIE*, vol. 2670, p. 239-40, 1996.

[16] D. Zhong and S.F. Chang, "Spatio-temporal video search using the object based video representation", *Proc. of ICIP*, vol. 1, p. 21-24, 1997.