Mosaic-Based Video Object Segmentation

Liang-Hua Chen and Yu-Chun Lai
Department of Computer Science
and Information Engineering
Fu Jen University, Taipei, Taiwan

Abstract: Object segmentation is a problem within the scope of MPEG-4 standardization activities. This paper proposes a novel technique to extract the moving objects in the video sequences. Our approach is based on the integration of mosaic-based temporal segmentation and color-based spatial segmentation. The mosaic representation of video allows us to fully exploit the spatio-temporal information in the video scene to achieve robust segmentation. Compared with the related works which detect motion by the difference of two consecutive frames, our approach uses the information aggregated over a group of frames. Thus, our system is more robust and is able to extract the non-rigid object that has complex motion from one frame to the next.

Key-Words: Video sequence analysis, image mosaics, object segmentation, motion detection, spatial segmentation.

1 Introduction

After over 30 years of video coding research and development, standards are moving towards a more general and powerful way of coding visual content, in which data is described in terms of ``objects". In the object-based standard, such as MPEG-4, the coder may perform a locally defined preprocessing, aimed at the automatic identification of the objects appearing in the video sequence. The video object may be a background or a moving object. Then, each extracted object is coded in a separate data stream. Therefore, object segmentation is a key issue in efficiently applying the MPEG-4 coding scheme. Apart from video coding, the technique of object segmentation can be applied to other areas such as surveillance, targeting and gesture recognition. In our work, we address the extraction of moving object from stationary background in a video sequence.

Segmentation techniques can be coarsely classified into spatial and temporal (motion) segmentation. Spatial segmentation provides accurate region boundaries, but a semantically meaningful object extraction cannot be achieved without further information. On the other hand, temporal segmentation can identify moving objects since most moving objects have distinct motion patterns from the background. However, it does not provides as accurate object boundaries as can be obtained from spatial segmentation. Because of these limitations, there has been a trend towards combining the spatial and temporal segmentations to obtain better segmentation result[1,2,3]. However, satisfactory results were reported for a certain type of video content, e.g., those with rigid objects and simple motions. The main reason is that these approaches do not fully exploit the spatial and temporal coherence information inherent in the video sequence. For example, most temporal segmentation methods take differences of intensity values between two consecutive image frames to find moving objects in space. It is well known that temporal gradient between two frames is sensitive to light variations, noise, camera motion and object motion. To achieve robust segmentation, in this paper, we propose a segmentation method which is based on information integration over a group of frames rather than two consecutive frames.

Recently, there has been a growing interest in the use of a *mosaic* image as an efficient way to represent a collection of frames [4]. A mosaic is a panoramic image obtained by aligning all images of a video sequence onto a common reference frame. This representation allows us to capture the spatio-temporal information contained in video sequences. Thus, in our approach, the mosaic-based temporal segmentation coarsely localizes the moving parts of an object and the color-based spatial segmentation divides the image into homogeneous regions with precise object boundaries. Finally, by integrating spatial and temporal segmentation results, the moving object regions are separated from the background accurately.

2 Mosaic Construction

Our mosaic-based motion segmentation method is carried out by comparing each frame with a mosaic of the static background. To build a static mosaic of the background, one must be able to align frames from a video shot. Video shot is a sequence of frames

continuously captured by the same camera. In our system, the video shots are obtained by applying our shot boundary detection algorithm [9] to the original video sequence.

We first need to derive the transformation between partially overlapping frames. Assuming background motion (due to camera motion) is the dominant motion of the video, then the image motion of the majority of scene points can be approximated by the following transformation model:

$$u(x, y) = a_1 x + a_2 y + a_3$$

$$v(x, y) = -a_2 x + a_1 y + a_4$$

where $(u(x, y), v(x, y))$ is the motion vector at image position (x, y). This model is a special case of affine model with 6 parameters. However, experimental results show that our model is better than the general affine model. This is because we make more constraints on the model parameters to avoid some undesirable transformations implied in affine model such as skewing. In this step, we need to obtain the motion vectors (or displacements) between successive frames. Previous approaches are based on the feature matching or optical flow computation. These techniques are computationally intensive. To reduce the processing time, we directly use the motion vectors encoded in the MPEG-1 video stream[5]. Fig. 1 shows one frame of a baseball game sequence and its corresponding motion vectors.

Given the motion vector, the image motion parameters can be estimated by a robust regression technique called least-median-squares[6]. Let the motion vectors of each frame be denoted as $\{[u_1(x_1, y_1), v_1(x_1, y_1)], \cdots, [u_n(x_n, y_n), v_n(x_n, y_n)]\}$. The least-median-squares method can be described as

$$\min_{\hat{a}} \{ \underset{1 \le i \le n}{median} [(a_1 x_i + a_2 y_i + a_3 - u_i)^2 + (-a_2 x_i + a_1 y_i + a_4 - v_i)^2]\}$$

where $\hat{a} = (a_1, a_2, a_3, a_4)$ is the parameter vector. One distinctive property of the algorithm is that it can tolerate up to 50% outliers in the data set, i.e., half of the data set can be arbitrary without significantly effecting the regression result. Therefore, this technique can robustly estimate the motion of the majority of scene points (background) and will not be biased by the minority scene points (moving object).

Once the transformations between successive frames have been determined, these transformations can be composed to obtain the alignment between any two frames of the video sequence, and in particular, between the current frame and the reference frame. In most cases, we choose the first frame of the sequence as a reference. After all frames have been aligned to a reference frame, the next step is the selection of pixels to be put into the resulting mosaic. The gray value in each pixel of the mosaic will be computed by applying an appropriate temporal operator to the aligned frames. The temporal average operator is effective in removing temporal noise, but the moving objects will appear blurred, with `ghost-like' traces in the resulting mosaic. The temporal median operator can remove the moving objects and produce a static mosaic of background, but it is computationally expensive and is an off-line process. Therefore, we propose a novel scheme which is both effective in deleting the moving objects and feasible for the on-line creating panoramic image. Our approach is based on the observation that each overlapping pixel of the aligned frames may fall into one of the two categories: background or moving object. Since background motion is the dominant motion of video and we want to build the mosaic of background, we select the pixel which appears the most frequently in the temporal domain. As illustrated in Fig. 3, there are some artifacts in the resulting mosaic. However, the effect of these artifacts can be reduced by some post-processing as described at the end of next section.

3 Temporal Segmentation

After the mosaic is constructed, moving objects are segmented out by computing the intensity difference between the current frame (after alignment) and the background mosaic. In our implementation, for each pixel, we estimate the average difference around a small neighborhood. The resulting image is thresholded to obtain a binary map. This binary map contains the blobs (connected components) produced by the moving objects and other small blobs due to misalignments or noise. To extract only relevant moving objects from the map, the following post-processing is needed. A size filter is effective in eliminating noisy areas in binary map. Any blob smaller than some threshold size is removed. Next, we use the morphological operator *closing* to produce more compact regions without altering the original shapes.

4 Spatial Segmentation

Spatial segmentation is the process that divides the image into homogeneous regions using the color

information at each pixel. To perform this task, one classic method consists of finding clusters of points in the 3D color space and labeling each cluster as a different region[7]. The main disadvantage of this method is the number of clusters (regions) is typically unknown for traditional data clustering algorithm such as k-means. Another problem with clustering is the spatial information is not taken into account. In this paper, we employ a new clustering algorithm called mean-shift [8] to determine the number of dominant colors automatically.

For each frame, dominant colors are first generated by the mean shift algorithm. Then, we explore the spatial relation of pixels to get the spatial segmentation result. All pixels are classified according to their distance to dominant colors in color space and spatial relationships within image domain. In our implementation, the color space CIE($L^*u^*v^*$) Is used. For each pixel, we assign it to the class with the shortest distance if the distance is smaller than a threshold. Afterwards, the threshold is increased by a certain amount. For each unassigned pixel, we assign it to a certain class if its distance to the corresponding dominant color is smaller than the modified threshold and one of its neighboring pixels has been assigned to the same class. Finally, all remaining unassigned pixels are classified to its nearest neighboring region. It is noted that the dominant colors of the current frame can be used as the initial guess of dominant colors in the next frame. Due to the similarity of adjacent frames, the mean shift algorithm often converges in one or two iterations. Thus, the computational time is reduced significantly.

5 Integration of Spatio-Temporal Segmentation

Because we take severe threshold value, temporal segmentation provides us with an object mask which is generally smaller in size than the true size of moving object. The spatial segmentation produces a number of homogeneous regions $R = \{R_1, \cdots, R_n\}$, and the object itself is composed of several regions. To integrate spatial and temporal segmentation results, each region R_i is examined to decide if it belongs to the moving object. When 20% of R_i is covered by the object mask, the whole region of R_i is declared as part of the moving object; otherwise the whole region of R_i is declared as the background. The resulting image is a labeled image indicating moving object and background.

6 Experimental Results

The proposed algorithm has been applied successfully to several test sequences. Here, we demonstrate the performance on "skater" and "soccer-player" sequences. Both have fast and complex motions. In Fig. 2, the original 20th, 40th and 60th frames are shown. The mosaic of the background is shown in Fig. 3. Fig. 4(a) shows the temporal segmentation result by using a thresholding difference between the 4th frame and the mosaic of background. The result of color segmentation for the 4th frame is shown in Fig. 4(b). After integrating the spatial and temporal segmentation, the extracted object is shown in Fig. 4(c). Fig. 5 shows some selected frames of "soccer-player" sequence. Each frame of this sequence contains multiple moving objects. The constructed mosaic and segmentation results are shown in Fig. 6 and Fig. 7, respectively.

To have an objective evaluation of the quality of our segmentation result, a spatial accuracy measure is calculated

$$S_{error} = \frac{\sum_{(x,y)} A^{ext}(x,y) \oplus A^{ref}(x,y)}{\sum_{(x,y)} A^{ref}(x,y)}$$

where A^{ext} and A^{ref} are binary masks for extracted object and reference object, respectively, and $\oplus$ is the binary *XOR* operation. Currently, the reference object is obtained by manual segmentation. The averaged error for "skater" and "soccer-player" sequences are 1.1% and 5.4%, respectively. The latter is a little bit high. Because, in some frames, two moving objects overlap each other, our current approach can not separate them completely.

7 Conclusion

We have presented a mosaic-based approach for the video object segmentation. The mosaic representation of video allows us to fully exploit the spatio-temporal information in the video scene to achieve robust segmentation. Other advantages of the proposed technique include

(1) A robust regression method based on the minimization of median-of-squares-errors between the data and fitted model is used to estimate camera motion. This method yields the correct result even when half of the data is incorrect to an arbitrary degree.

(2) A novel "most-frequently-appear" strategy is proposed to integrate the aligned frames to form mosaic image. It is not only computationally

inexpensive but also effective in removing the moving objects.

(3) The temporal segmentation is based on the information aggregated over a group of frames rather than two consecutive frames.

(4) The object segmentation algorithm combines spatial and temporal segmentations in a synergetic manner to complement their respective deficiencies.

While most previous approaches only work on object with simple motion, our approach can accurately extract object with fast and complex motion. Experimental results show that the proposed technique is effective and can be applied to object-based video coding. Finally, our future work should also be directed towards the extraction of multiple objects which overlap each other.

References:

[1] S. Herrmann, H. Mooshofer, H. Dietrich, and W. Stechele. A video segmentation algorithm for hierarchical object representations and its implementation. *IEEE Trans. Circuits Syst. Video Technol.*, 9(8):1204-1215, Dec. 1999.

[2] M. Kim et. al. A VOP generation tool: automatic segmentation of moving objects in image sequences based on spatial-temporal information. *IEEE Trans. Circuits Syst. Video Technol.*, 9(8):1216-1226, Dec. 1999.

[3] I. Patras, E.A. Hendriks, and R.L. Lagendijk. Video segmentation by MAP labeling of watershed segments. *IEEE Trans. on Pattern Anal. Machine Intell.*, 23(3):326-332, March 2001.

[4] M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu. Efficient representations of video sequences and their applications. *Signal Processing: Image Commun.*, 8(4):327-351, May 1996.

[5] D.L. Gall. MPEG: a video compression standard for multimedia applications. *Communication of ACM*, 34(4):46-58, April 1991.

[6] P.J. Rousseeuw and A.M. Leroy. *Robust Regression & Outliers Detection*. John Wiley & Sons, New York, 1987.

[7] Q. Ye, W. Gao, and W. Zeng. Color image segmentation using density-based clustering. In *Proc. IEEE Int. Conf. Acoustic, Speech and Signal Processing*, volume 3, pages 345-348, 2003.

[8] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(5):603-619, May 2002.

[9] L.-H. Chen, C.-W. Su, H.-Y. Liao, and C.-C. Shih. On the preview of digital movies. *Journal of Visual Communication and Image Representation*, 14(3):357-367, September 2003.

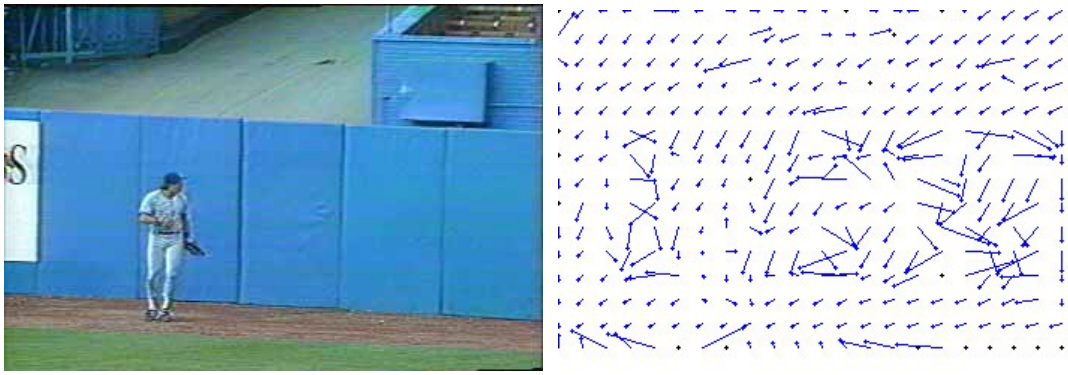
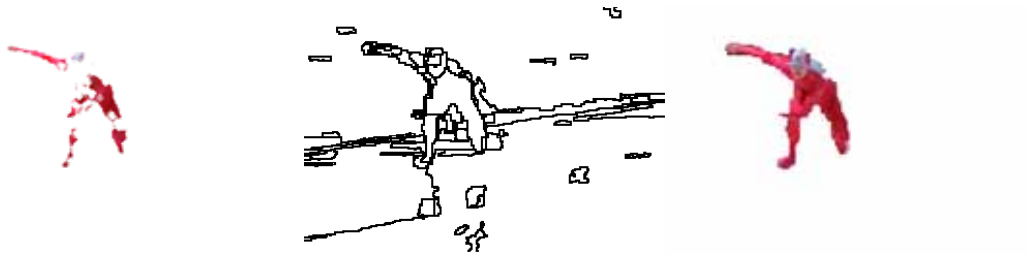


Fig. 1 : One frame of a baseball game sequence and its corresponding motion vectors.

Fig. 2 : The 20th , 40th and 60th frames of “Skater” sequence.

Fig. 3 : The static mosaic of “Skater” sequence.



(a) (b) (c)

Fig. 4 : (a) is result of temporal segmentation , (b) is result of spatial segmentation , and (c) is the extracted object.

Fig. 5 : The 20^{th} , 60^{th} and 100^{th} frames of “soccer-player” sequence.

Fig. 6 : The static mosaic of “soccer-player” sequence.

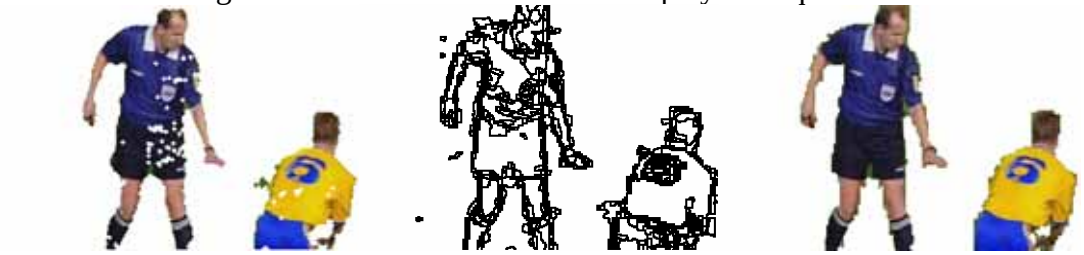


Fig. 7 : (a) is result of temporal segmentation , (b) is result of spatial segmentation , and (c) is the extracted object.