SUPERVISED VIDEO OBJECT SEGMENTATION USING A SMALL NUMBER OF INTERACTIONS

Hanfeng Chen[1] *Feihu Qi*[1] *Su Zhang*[2]

[1]Dept. of Computer Science & Engineering
[2]Institute of Biomedical Instrument
Univ. of Shanghai Jiao Tong, Shanghai, 200030, China

ABSTRACT

In this paper, a supervised video object segmentation algorithm using a small number of interactions is proposed. The proposed algorithm is composed of three steps: semi-automatic first frame segmentation, automatic object tracking and boundary refinement. Homogeneous region segmentation is performed before the user interaction for the first frame to minimize the amount of interaction. Then, a polygon with very few key nodes can be drawn conveniently to get the segmentation mask of the first frame. In the object region tracking, pixel-wised backward tracking is adopted. Finally, a method for the mask refinement is proposed by considering similar pixels of each pixel in its neighbor region. Extensive experimental results show that the proposed algorithm is effective for video object segmentation semi-automatically.

1. INTRODUCTION

Video semantic object segmentation is crucial to realize the content-based concept emphasized in MPEG-4 and MPEG-7. Video semantic object segmentation means to segment interested semantic object layer from a video sequence. Currently developed algorithms for semantic object segmentation include two categories: unsupervised and supervised segmentation.

Unsupervised segmentation[1-3] is attractive in real-time systems. However, unsupervised segmentation does not work in many cases. Only such simple semantic objects as moving objects can be defined in the unsupervised segmentation. A moving camera may also bring troubles to the segmentation. Moreover, boundaries of some spatially neighbored objects are hard to be segmented in unsupervised segmentation if these objects are of the same color and texture. In this paper, we will focus on supervised segmentation.

In the supervised segmentation[4-9], user interaction is added to the segmentation process. By user interaction a semantic object can be defined easily and some uncertain borders can be decided. User interaction happens usually at the initial stage of the segmentation, such as the first frame, and a mask is gotten to indicate the object region. Then, the object region is tracked in the rest frames. The mask of each frame is refined during the tracking.

What should be emphasized in the supervised segmentation is that the user interaction does not mean to supersede the dominant function of the computer. The amount of user interaction should be minimized to improve the efficiency of the supervised segmentation, which is neglected in many previous methods. In those methods[6,8], two approximate contours near the true contour of the object to be segmented are drawn out manually. One is out of the true object region and the other one is in the object region. Then, certain algorithms, such as watershed method, are performed to approach the two contours to the true contour. It is a burdened job to draw the approximate contours, especially when the shape of the object is complex. The second stage of the supervised segmentation is to track the object region in the rest frames automatically[6,7]. The most commonly used methods for tracking are region tracking and contour tracking. These two methods are based on automatic homogeneous regions segmentation or edges detection in the rest frames. However, automatic homogeneous regions segmentation or edges detection are not so reliable sometimes, especially when neighbored regions are of similar low-level features.

In this paper, a novel supervised object segmentation and tracking algorithm is proposed. The proposed algorithm also adopts the three-steps configuration of segmenting first frame semi-automatically, tracking in the rest frames and refinement. The main differences between the proposed method and the previous lie in the next three points. Only a few keystrokes are needed for the semi-automatic first frame segmentation with the assistance of automatic gray-homogeneous or texture-homogeneous regions segmentation and these keystrokes are not required to cling to the object boundary closely. Backward tracking based on block-matching which is

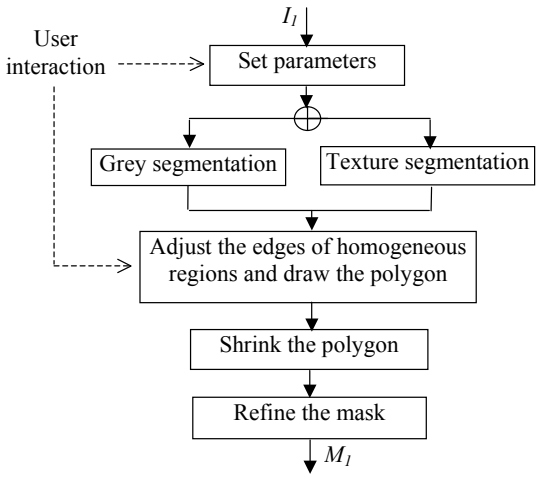

Fig.1 Flow of first frame segmentation

performed only to the boundary regions of previous segmentation mask is adopted in the segmentation of the rest frames. Finally, a simple but effective algorithm for the refinement is proposed.

2. FIRST FRAME SEGMENTATION

In many previous methods, one or two approximate contours near the true contour of the object to be segmented are drawn out manually when segmenting object region in the first frame. Then, certain algorithms, such as watershed method, are performed to approach the approximate contours to the true contour. It is a burdened job to press and move the mouse to cling to the object contour, especially when the shape of the object is complex. A new method for the first frame segmentation is presented to decrease the amount of user interaction in this paper. Fig.1 describes the flow of the method.

Firstly, some parameters necessary in the algorithm are set manually according to different scenes. Next, the first frame I_1 is segmented into various homogeneous regions, illustrated as Fig.2. For usual scenes an image segmentation method based on mean shift algorithm[10] is performed to segment the first frame into gray-homogeneous regions. The improved image segmentation algorithm can eliminate the noise and rich microgrooves in the image. For those scenes with rich texture regions a texture segmentation algorithm[11] is adopted to segment the image into various texture-homogeneous regions. This texture segmentation algorithm works well in getting edges of texture-homogeneous regions. Nevertheless, a few neighbored regions may be merged in the segmentation because of similar gray or texture. These merged regions are sheared break by moving the mouse, illustrated as the short overstriking curve in Fig.2. In most cases, the amount of shear is very small.

Then, a user needs to provide a few key nodes orderly and a polygon illustrated as the broken line in

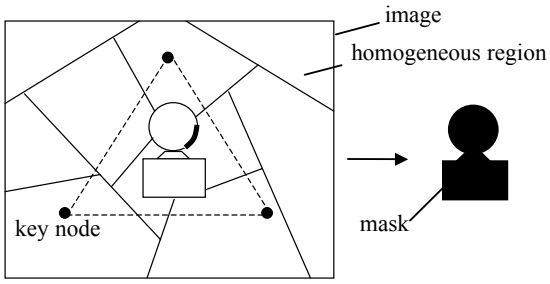

Fig.2 First frame segmentation

Fig.2 is gotten accordingly. The polygon should contain the whole needed object region and traverse all the homogeneous regions neighbored with the object region. Very few keystrokes are needed and the locations of the keystrokes are freer because the key nodes are not required to cling to the contour of the object region. So, the amount of interaction is decreased significantly compared with drawing the approximate contours.

Finally, the polygon shrinks along homogeneous regions to the edges between the object region and its neighbored homogeneous regions. The final location of the polygon shows the object region of the segmentation mask M_1. A refinement process which will be given out in section 4 is performed to refine the mask M_1.

3. BACKWARD TRACKING

Let M_{n-1} the segmentation mask of the previous frame I_{n-1}. The segmentation mask M_n of current frame I_n is gotten by backward tracking. M_{n-1} and M_n are assumed to be similar in location and shape. This assumption is true in most cases. Thus, backward tracking is performed only to the *tracking region* T_{n-1} but not all the pixels. In this paper *tracking region* T_{n-1} means those pixels around the boundary of M_{n-1}. T_{n-1} can be decided by simple morphological 'dilate' operation on the boundary of M_{n-1}, illustrated as Fig.3. How much times the 'dilate' is applied can be decided according to the speed of the moving object. Five times are enough usually.

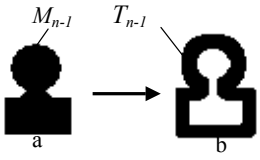

Fig.3 Get *tracking region*

The backward tracking is done by block-matching. Block-matching is deficient in modeling the real movement accurately. However, it can approximate the local movement of pixels in most cases and the approximation is sufficient in this application. Let $p_n(i,j)$ a pixel of I_n locating in T_{n-1} and $p_{n-1}(m,n)$ is its matching pixel in I_{n-1}. $p_n(i,j)$ is considered to belong to the object region of M_n if $p_{n-1}(m,n)$ belong to that of M_{n-1}. Moreover, the residual region R_{n-1} of M_{n-1} subtracted by T_{n-1} is considered to belong to the object region of M_n inherently.

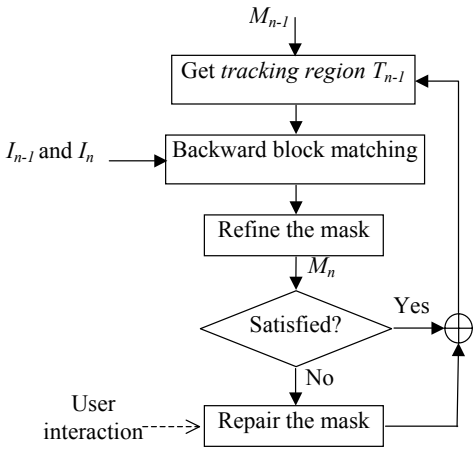

Fig.4 Flow of backward tracking

Then, the gotten mask is refined as the first frame done. Fig.4 shows the flow of the tracking process.

If the mask is unsatisfied, a user has two selections. One is to set the current frame as the first frame and restart the semi-automatic segmentation. The other one is to repair the mask partly with a mouse 'paster'. A paster can select an arbitrary rectangular region of the mask, object region or background region, and pastes the selected region on any location of the mask.

4. REFINEMENT OF SEGMENTATION MASKS

A segmentation mask of current frame has been gotten with the backward tracking. There may be some inaccuracies around the boundary of the mask. Some parts of the real object region are missed in the segmentation mask and some parts of the segmentation mask do not belong to the real object region. Fortunately, these inaccuracies concentrate on the regions around the boundary of the mask and the inaccurate regions are tiny compared with the whole mask. However, these inaccuracies should not be ignored because they will be cumulated in the tracking of the rest frames. A simple algorithm is presented in our paper to refine the segmentation mask.

Fig.5 shows a microcosmic part around the object boundary, where pixels signed with triangles belong to the real object region and the others belong to the background. The real curve is the boundary of the segmentation mask before refinement. The refinement is performed on the exterior boundary pixels and interior boundary pixels. Here, exterior boundary pixels are those neighboring to the object boundary and belonging to the background and interior boundary pixels are those neighboring to the object boundary and belonging to the object.

Let *a* be an interior boundary pixel and A the set of its N×N neighbor pixels. N is decided on the smoothness of the scene and set as 7 in this paper. The distances

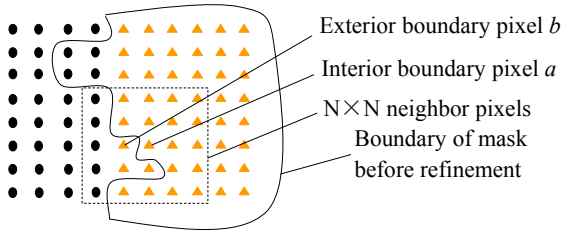

Fig.5 A microcosmic part around the object boundary (N=5)

between *a* and pixels in A are measured. The absolute difference of the gray values is adopted to define the distance of two pixels in our paper. Then, these distances are sorted and a ratio of pixels of A which have smallest distances with *a* are extracted into set A'. The ratio is set as 0.6 in our work. Let

$$\rho = \frac{S_1}{S} \tag{1}$$

where S is the number of pixels in A' and S_1 the number of pixels signed to be object pixel in A'. Then, the pixel *a* is decided according to the following rule

$$mask(a) = \begin{cases} 1 & if\ (\rho \geq T) \\ 0 & if\ (\rho < T) \end{cases} \tag{2}$$

where the fact $mask(a)$ equals to one means that *a* belongs to the object region and zero means that *a* belongs to the background. T is a threshold and set as 0.4 in our experiments.

Each exterior boundary pixel can be decided on the same rule as (2) with the only difference that T should be replaced by 1-T. So, the process of the refinement can be described as followed:

1. Let C_{in} be the set of all the interior boundary pixels of the segmentation mask. Each pixel of C_{in} is decided according to the rule (2). Then, C_{in} is updated according to the modified mask.
2. Repeat step 1 for *m* times.
3. Let C_{out} be the set of the exterior boundary pixels of the modified mask after step2. Each pixel of C_{out} is decided according to rule (2) and then updates C_{out}.
4. Repeat step 3 for *m* times.

Usually, such a small value of *m* as two or three is enough for the refinement since the mistakes are tiny. Thus, the algorithm can be performed rapidly. Another advantage of the algorithm is that it will not destroy the boundary greatly when the boundary is crossing a large homogeneous region, even if a large *m*. It is because ρ approximates to 0.5 in this case.

5. EXPERIMENTAL RESULTS

Several MPEG-4 test sequences are segmented in our experiments. Fig.6-9 show some of the experimental results. In these shown results, user interactions are happened only to the first frames and there is no further user assistance in the remaining frames. For the *Mother &*

Daughter sequence, gray-homogeneous region segmentation is performed to the first frame before the polygon is drawn manually, shown as Fig.6(b). Similar processes are followed for segmenting sequences *Foreman* and *Missa*. For the Tennis sequence, texture–homogeneous region segmentation is performed to the first frame because the background is homogeneous in texture, shown as Fig.7(b) where each texture-homogeneous region is rimmed with white edges. It can be found in Fig.6(b) and Fig.7(b) that only very few key nodes are needed to construct the polygon. Though the user interactions are decreased, the tracking results are consistent in a long period of frames.

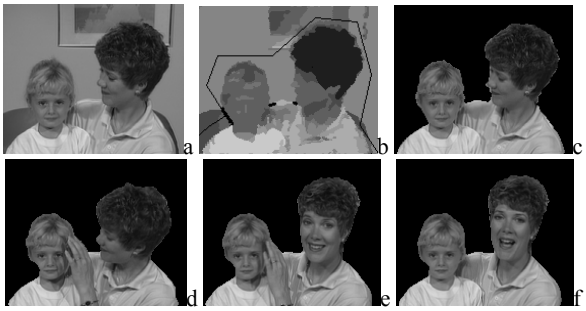

Fig.6 Results from the *Mother & daughter* sequence: (a) original frame, (b) gray-homogeneous regions segmented automatically and the polygon inputted manually, and (c) segmented 1th, (d) 30th, (e) 50th, and (f) 70th frames.

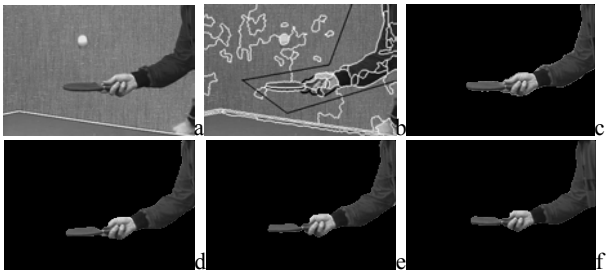

Fig.7 Results from the *Tennis* sequence: (a) original frame, (b) texture-homogeneous regions segmented automatically and the polygon input manually, and (c) segmented 1th, (d) 10 th, (e) 20 th, and (f) 30 th frames.

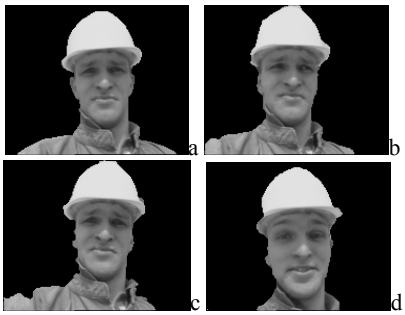

Fig.8 Results from the *Foreman* sequence: (a) 1th frame segmented semi-automatically, and (b) segmented automatically 30th, (c) 50th, and (d) 100th frames.

Fig.9 Results from the *Missa* sequence: (a) 1th frame segmented semi-automatically, and (b) segmented automatically 30th, (c) 60th, and (d) 90th frames.

6. ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China (No. 60072029).

7. REFERENCES

[1] Meier, T. and Ngan, K.N., "Automatic Segmentation of Moving Objects for Video Object Plane Generation", IEEE Transactions on Circuits and Systems for Video Technology, vol.8, No.5, pp.525-538, 1998.

[2] A. Neri, S. Colonnese, G. Russo and C. Tabacco, "Adaptive Segmentation of Moving Object Versus Background for Video Coding", SPIE 1997, vol.3164, pp.443-453, 1997.

[3] D. Zhong and S.-F. Chang, "Long-Term Moving Object Segmentation and Tracking Using Spatio-Temporal Consistency", ICIP01, vol.2, pp.57-60, 2001.

[4] M.Kim, J.G., M.H.Lee and C.Ahn, "User-assisted Segmentation for Moving Objects of Interest", Doc. ISO/IEC JTC1/SC29/WG11 M2803, July 1997.

[5] D. Zhong and S.-Fu Chang, "AMOS-An Active MPEG-4 Video Object Segmentation System", ICIP98, vol.2, pp.647-651, 1998.

[6] Chuang Gu and Ming-Chieh Lee, "Semiautomatic Segmentation and Tracking of Semantic Video Objects", IEEE Transactions on Circuits and Systems for Video Technology, vol. 8, No. 5, pp. 572-584, 1998.

[7] Ju Guo, Jongwon Kim and Kuo C.-C.J., "An Interactive Object Segmentation System for MPEG Video", Proceedings of ICIP 99, vol.2, pp.140-144, 1999.

[8] I. Grinias and G. Tziritas, "A Semi-automatic Seeded Region Growing Algorithm for Video Object Localization and Tracking", Signal Processing: Image Communication, vol.16, pp.977-986, 2001.

[9] Jungeun Lim and Jong Beom Ra, "Semi-automatic Video Segmentation for Object Tracking", ICIP01, vol.2, pp.8-84, 2001.

[10] D. Comaniciu, V. Ramesh and P. Meer, "The variable bandwidth mean shift and data-driven scale selection", ICCV01, vol.1, pp.438-445, 2001.

[11] Yining Deng and B.S. Manjunath, "Unsupervised Segmentation of Color-Texture Regions in Images and Video", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, No. 8, pp.800-810, 2001.