FAST TWO-STEP UNASSISTED VIDEO SEGMENTATION TECHNIQUE EVALUATED BY TOLERANT GROUND TRUTH

Piotr Steć[1], *Marek Domański*[2]

[1] University of Zielona Góra, Institute of Control and Computation Engineering, Zielona Góra, Poland, email: P.Stec@issi.uz.zgora.pl

[2] Poznań University of Technology, Institute of Electronics and Telecommunications, Piotrowo 3A, 60-965, Poznań, Poland, e-mail: domanski @et.put.poznan.pl

ABSTRACT

This paper presents a fast unassisted video segmentation technique. It exploits high performance of the Fast Marching Method. In contrary to the typical approach an active contour is initialized as a point inside a frame rather then as a border of a frame. The technique can be extended to segmentation methods which allow soft edges of the segment. Additionally, a new quality evaluation technique is proposed which gives results similar to human assertion.

1. INTRODUCTION

The video segmentation is essential for object-based media delivery. Standards like MPEG-4 and MPEG-7 [1, 2] refer to techniques and methods that are able to handle different objects separately, do not define segmentation procedures. This is left up to user and is subject of intensive research recently. The most challenging task, and simultaneously the topic of the work, is real-time unassisted video segmentation.

For the sake of brevity, the review of the existing approaches should be left beyond the scope of the paper. Among various approaches, active contours are often used for image segmentation and object extraction. Here, the approach based on active contours is used in order to segment independently frames from a video sequence.

Here, the goal is to exploit both motion and color information available in a video sequence. It is possible to detect a semantic object in a video sequence thanks to its motion consistency. However in practice it is very difficult task, because estimation of dense and highly accurate motion field is a still unsolved problem. The proposed method is able to perform even on erroneous motion field produced by the known dense motion estimation algorithms.

2. INITIALIZATION PROCEDURE

Typically, video segmentation algorithms based on active contour methods are initialized at border pixels of a frame. This is very convenient assumption but objects that are partially visible in the frame may be not detected. This happens because initial contour intersects such object as well as the background.

Application of Fast Marching Method [3, 4, 5] makes it possible to start at any point of background and segment out object thanks to possibility of contour topology change. It is even possible to make foreground-background segmentation in presence of multiple moving objects.

The serious problem is to automatically find a good initialization point. The initialization procedure based on displaced frame difference estimation will be proposed. It is likely possible that the largest, uniformly moving region will be the background region. The difference at the point (x,y) is computed as follows:

$$dfd = \left\langle \sum_{i=-1}^{1}\sum_{j=-1}^{1} \left| I_n(x+i, y+i) - I_{n-1}(mx+i, my+j) \right| \right\rangle, \quad (1)$$

where mx and my are the position of the point in the previous frame according to the motion information and $\langle . \rangle$ denotes mean. Such computation procedure has two purposes: reduction of the image noise influence and reject single points as starting region candidates. All the connected pixels with the dfd value equal to zero are connected as one region. As the initialization point the center point of the largest region is chosen.

3. SEGMENTATION ALGORITHM

In order to exploit speed and elasticity of Fast Marching Method and to avoid its week points at the same time, a two step algorithm was developed. This technique was described in its preliminary version in [6]. Here, some other aspects and extensions are considered.

In the first step, a rough segmentation is performed using Fast Marching Method. In the second step segmentation results are refined using static image segmentation algorithm.

3.1. Rough segmentation

In the initial segmentation, the most important part was proper definition of the contour propagation speed F for the propagation equation:

$$|\nabla T|F = 1 \quad (2)$$

The speed function should be high in the areas with uniform colour to provide insensitivity to motion estimation errors which are common in uniform image areas. At the same time speed must be high in the areas of smooth motion. It allows a contour to pass highly textured image areas, where motion estimation is more accurate. The value of the speed function drops towards zero in the areas where strong edges in both colour and motion field exist.

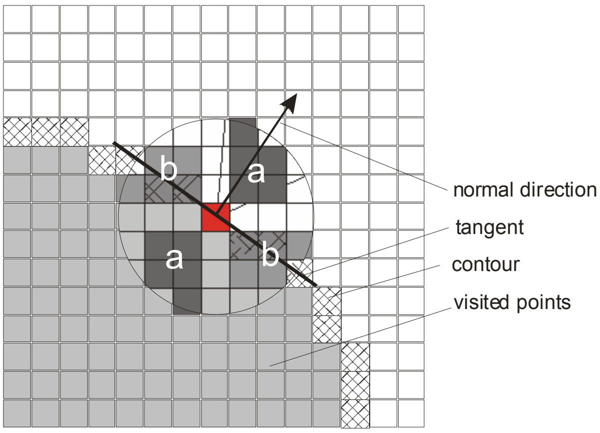

Fig. 1. Curve-oriented difference operator.

The speed function is defined as follows:

$$F = \frac{1}{\min(dI, \alpha \cdot dM) + 1} \cdot k\,, \quad (3)$$

where dI is the difference in colour, dM is the difference in motion and k is the curvature-dependent smoothing term. The parameter α is the normalizing coefficient which scales the motion difference to the range comparable with image difference. This parameter can be computed by finding the maximum difference in motion field and by scaling it to the maximum difference in image.

The propagation of the contour should stop before real object boundary is reached. This is very important while contour propagated by the Fast Marching algorithm is not able to move back and correct possible error. If object boundary is missed, the contours leaks into object making impossible detection of object or it major part can be missed. It often happens when contour approaches object boundary from a direction being square to the boundary. To reduce this problem, a curve-oriented operator was designed (Fig. 1). This operator is always aligned to the normal direction of the propagating contour. Its "b" parts (Fig. 1) are responsible for slowing down parts of the contour which are perpendicular to actual object boundary. This forces the contour to turn towards object boundary. Thanks to it, real object edges are easier to detect by the "a" parts of the operator.

The edges should be detected soon enough to prevent contour from leaking into an object. In the perfect case, speed should drop to zero near object edge and propagation should stop at that point. However, in practice, a threshold must be applied. Propagation of the contour stops at points, where speed is lower then the threshold.

The threshold can be adjusted automatically according to the following procedure.

$$F_{\min} = \frac{1}{\min(dI_{\max}, \alpha \cdot dM_{\max}) + 1}\,, \quad (4)$$

$$F_{th} = F_{\min} + \left(1 - F_{\min}\right) \cdot \alpha\,, \quad (5)$$

where dI_{max} and dM_{max} are the maximum difference in image and the maximum difference in motion, respectively. $F_{\min}$ is the minimum possible speed. The parameter α is chosen empirically on the basis of the typical speed variation in a number of sequences. Its value is set to 0.1. It means that the threshold is set to 10% of the speed range possible for a considered image.

3.2. Refinement step

The first step of the algorithm stops the propagating contour before object boundary. It is necessary, because Fast Marching Method cannot move a contour backwards. If the object boundary was passed by propagating curve at any point, the curve would leak into the object linking it to background.

In the second step of the algorithm, segment borders are moved towards nearest edge detected in the picture by static image segmentation. Segmentation of the whole image would be very time-consuming, and it is not necessary. Only a part that is contained in the band around curve developed in the first stage is considered. Thickness of the segmentation area depends on the static segmentation method used. Generally it should be wider then average segment produced by static segmentation.

Let sr_i be a i-th region obtained from static segmentation and fmr is a region obtained by Fast Marching segmentation.

for i=0 to n

if $sr_i \cap fmr \neq \emptyset$ add sr_i to fmr

where n is a total number of segments obtained from static segmentation. Above procedure merges segments certain segments obtained from static segmentation with those obtained in the first step. The segments with the common parts are merged.

4. SEGMENTATION ACCURACY EVALUATION

Segmentation quality assessment is an important part of video segmentation problem. The problem is that it is very difficult to obtain an objective quality measure that conforms with subjective assessments made by human observers.

In order to make segmentation results comparable, numerous methods were developed [7, 8, 9]. An obviously reasonable procedure is to compare evaluated segmentation to a ground truth using some kind of similarity measure. Despite of the similarity measure, there is a common problem with the definition of the reference segmentation. It is commonly assumed that the segmentation is binary. In the image anyway it is at least one pixel at object border with uncertain membership. Due to low sampling resolution and blurring by optics, this pixel value results from light from the object and that from the background. Problem even arises with the moving objects where, due to motion blur, uncertain boundary can be even thicker. The ground truth is defined by a human operator and such uncertain pixels can cause problems. Including or not the border pixels to the reference may cause very big errors in evaluation especially for objects with small area comparing to its circumference.

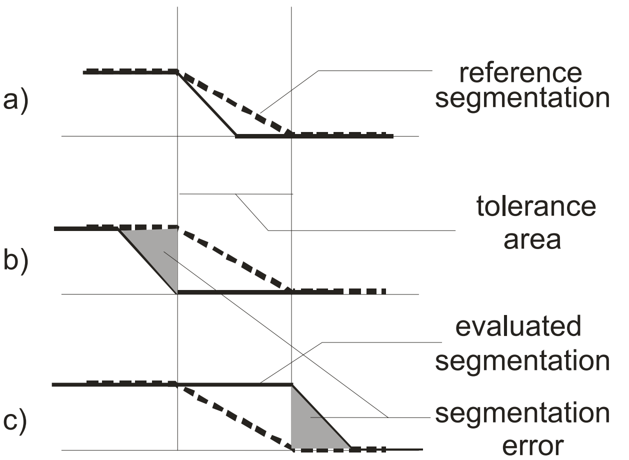

Fig. 2. Cross-section through the object edge. Areas that are erroneous in presence of tolerant ground-truth are marked with gray.

To overcome this problem a new evaluation procedure will be proposed. The ground truth must be prepared in different way than it is done typically. It can be assumed, that operator is able to determine which pixels belong to object for sure and which pixels belong to background. Namely two segments are prepared: object and background. Between these segments must be at last one pixel uncertainty area.

To be more general and to allow an extension to non-binary segmentation, considerations are made in continuous domain, i.e. they result in continuously valued segmentation masks. It is assumed that the area inside an object has value equal 1 and area outside the object value 0. The continuous-valued mask is obtained by interpolation. Lines (Fig. 2) are drawn between pixel centre points. This gives a line with angle equal to 45° for the binary segmentation. For more blurred segmentation, the angle will be lower. Figure 2 shows three possible positions of evaluated object edge (solid line). The absolute segmentation error is computed as a volume of differences between evaluated object and reference object computed outside tolerance area. This error value can be used as an input to some metric proposed in literature [7,9] to make it more comparable to object area.

5. EXPERIMENTAL RESULTS

The technique has been tested and evaluated for several video sequences. Two examples are shown in Fig. 3.

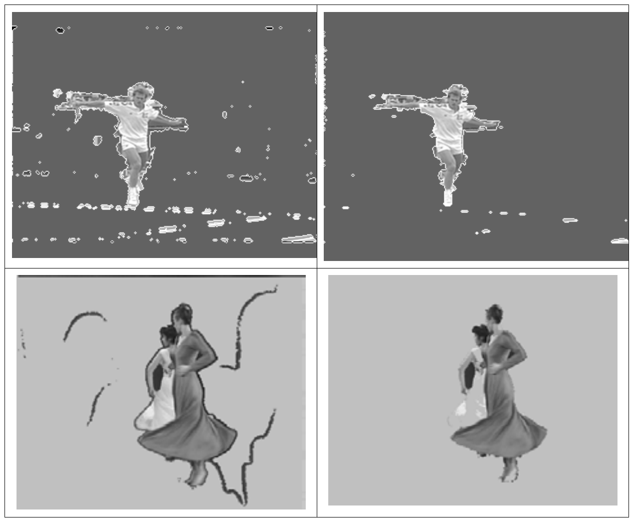

Fig. 3. First column shows segmentation after the first step, second after the refinement step.

For the first row ("stefan" sequence), the inaccuracy was 15% after the first step. The second step improved quality and only 3% of wrongly segmented pixels were left. The second row shows "dance" sequence with the results 30% and 4%, respectively. At the present stage of development, the algorithm needs about 1.5 second for segmentation of a frame using a 1.4 GHz AMD processor. However it is debug software version with significant testing and visualization overload. The complexity of the algorithm makes possible to get a real-time implementation in the near future.

6. CONCLUSIONS

Combination of two inaccurate methods but with different properties gave significant improvements in segmentation quality. Namely Fast Marching method for rough segmentation and simple static image segmentation algorithm for finding a nearest edge.

Proper construction of the speed function that moves segmenting contour allowed reduction of influence of motion estimation errors. Anyway, motion errors that appear near object edges can still decrease accuracy of segmentation. Better motion estimation algorithm would improve performance.

The technique is a good candidate for real-time tasks. Its computational complexity is low enough to allow such an implementation on modern equipment.

The new evaluation method provides assessments similar to those perceived by humans. Now, a small shift or scale, that is hardly perceived, does not generate substantial evaluation errors. Additionally, the method can be applied to soft-selected segments which will be probably the next stage of video segmentation research.

REFERENCES

[1] ISO/IEC IS 14496-2: Generic Coding of Audio-Visual Objects. Part 2: Visual.

[2] ISO/IEC CD 15938-3 "Information Technology – Multimedia Content Description Interface. Part 3: Visual".

[3] J.A. Sethian.: *Level Set Methods*. Cambridge University Press (1996).

[4] D. Adalsteinsson, J.A. Sethian: "A Fast Level Set Method for Propagating Interfaces." *Journal of Comp. Physics*, 118 (1995).

[5] J.A Sethian: "Fast marching methods." *SIAM Review*, vol. 41 (1999) no. 2, 199–235.

[6] P. Steć, M. Domański: Efficient Unassisted Video Segmentation Using Enhanced Fast Marching, IEEE International Conference on Image Processing, Barcelona, Spain 2003, vol. II, pp, 427-430.

[7] Ç. Erdem and B. Sankur, "Performance evaluation metrics for object-based video segmentation," in X European Signal Processing Conference (EUSIPCO), (Tampere, Finland), September 4-8 2000.

[8] P. Correia and F. Pereira, "Standalone objective evaluation of segmentation quality," in WIAMIS 2001 - Workshop on Image Analysis for Multimedia Interactive Services, (Tampere, Finland), 16-17 May 2001.

[9] Ç. Erdem, A. M.Tekalp, and B. Sankur, "Metrics for performance evaluation of video object segmentation and tracking without ground-truth," in International Conference on Image Processing (ICIP), (Thessaloniki, Greece), October 7-10 2001.