TEMPORAL STABILIZATION OF VIDEO OBJECT SEGMENTATION FOR 3D-TV APPLICATIONS

Çiğdem Eroğlu Erdem [1], *Fabian Ernst*[2], *Andre Redert* [2] *and Emile Hendriks* [3]

[1] Momentum A. Ş., İstanbul, Turkey
[2] Philips Research Laboratories, Eindhoven, The Netherlands
[3] Faculty of Electrical Engineering, Delft University of Technology, The Netherlands
E-mail: cigdem@ieee.org, {fabian.ernst,andre.redert}@philips.com, E.A.Hendriks@ewi.tudelft.nl

ABSTRACT

We present a method for improving the temporal stability of video object segmentation algorithms for 3D-TV applications. First, two quantitative measures to evaluate temporal stability without ground-truth are presented. Then, a pseudo-3D curve evolution method, which spatio-temporally stabilizes the estimated object segments is introduced. Temporal stability is achieved by re-distributing existing object segmentation errors such that they will be less disturbing when the scene is rendered and viewed in 3D. Our starting point is the hypothesis that if making segmentation errors are inevitable, they should be made in a temporally consistent way for 3D TV applications. This hypothesis is supported by the experiments, which show that there is significant improvement in segmentation quality both in terms of the objective quantitative measures and in terms of the viewing comfort in subjective perceptual tests. This shows that it is possible to increase the object segmentation quality without increasing the actual segmentation accuracy.

1. INTRODUCTION

The task of building 3D models of a time-varying scene, using the 2D views recorded by uncalibrated cameras is an important but unsolved task to provide content for the newly emerging 3D TV [1]. One approach to this problem is to segment the objects in the scene and order their video object planes (VOPs) with respect to their inferred relative depths. This approach gives a satisfactory sense of three dimensions when the scene is viewed in stereo. However, one of the most important requirements is the *temporal stability* of the video object planes. The changes in video due to occlusions, camera motion, changing background and noise should not cause sudden changes (temporal instabilities) in the *shape* and *color composition* of the video object planes (see Fig.1(c)), as they cause very disturbing flickering effects when the scene is viewed in stereo in 3D TV applications.

Many object segmentation and tracking algorithms exist in the literature [2]. These algorithms may loose temporal stability under difficult conditions, e.g. when the colors of the object and the background are similar causing missing object boundaries or when the motion can not be estimated with sufficient accuracy. In this paper we try to answer the question: "If making object segmentation errors are inevitable, how can we conceal them in our application?" Our approach is based on the hypothesis that if making segmentation errors are inevitable, they should be done in a temporally consistent way to increase the viewing comfort in 3D TV applications. To this effect, we propose a pseudo-3D curve evolution technique, which distributes the existing segmentation errors such that they will be less visible when the scene is rendered and viewed in stereo. The input to the proposed algorithm is a set of temporally unstable object segmentation maps which is estimated by any algorithm in the literature, for example by [3].

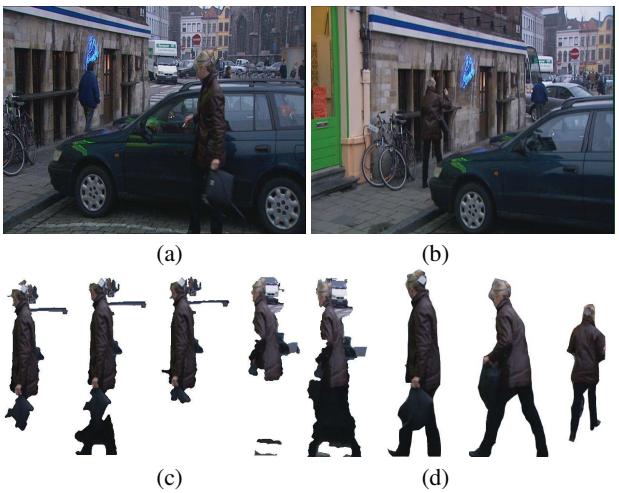


Fig. 1. (a), (b) First and last frames of "Flikken" sequence. (c) The given temporally unstable video object planes for the "lady" object (frames 8, 9, 10, 80, 81) from left to right. (d) Ground-truth VOPs for frames 8, 80 and 145.

2. MEASURES FOR TEMPORAL STABILITY

Assuming that the color histogram of the object does not change drastically from frame to frame, we can expect that a temporally stable object segmentation exhibits small differences between the color histograms of the estimated video object planes (VOPs) [4]. One shortcoming of the histogram measure is that it cannot distinguish if a portion of the object is removed and replaced by another block of the same color belonging to the background. Therefore, we can also require that the shape of two successive video object planes should not differ drastically. Hence, histogram and shape differences between successive video object planes are two candidates for evaluating the temporal stability of object segmentation.

Histogram Measure: The difference between two histograms can be calculated using the chi-square measure as follows [4]:

$$d_{\chi^2}(H_{t-1}, H_t) = \sum_{j=1}^{B} \frac{[H_{t-1}(j) - H_t(j)]^2}{H_{t-1}(j) + H_t(j)}, \qquad (1)$$

where H_t and H_{t-1} denote the RGB color histograms of the video object planes at frames t and $t-1$; B is the number of bins in the histogram. A prior normalization of the histograms may be necessary (see [4] for details).

Shape Measure: One way to represent the "shape" of a video object is to use the turning angle function of the boundary pixels [5]. The turning angle function (TAF) plots the counter clockwise angle from the x-axis as a function of the boundary length [5]. After obtaining the TAFs belonging to the video objects in successive frames, which are one dimensional vectors describing the shapes (denoted by θ_t and θ_{t-1}), the distance between them is calculated as follows:

$$d(\theta_{t-1}, \theta_t) = \frac{\sum_{j=1}^{K} \|\theta_{t-1}(j) - \theta_t(j)\|}{2\pi K} \quad (2)$$

where K is the total number of points on the boundary. In order for this function to be independent of rotation and of the choice of the starting point, the difference calculation (2) should be repeated after shifting one of the turning angle functions horizontally and vertically by increasing amounts, and then the minimum of the differences should be taken.

3. TEMPORAL STABILIZATION OF OBJECT SEGMENTATION MAPS

3.1. Background Theory

Region-based curve evolution techniques have been used for image segmentation in the literature [6, 7], where the region to be segmented can be characterized by a predetermined set of distinct features such as mean, variance, and texture, which may be inferred from the data.

A simple image segmentation problem is the case where there are just two types of regions in the image. Starting with an arbitrary initialization as denoted by $\overrightarrow{C}$, the curve $\overrightarrow{C}$ is evolved in such a way that it will eventually snap to the desired object boundary ∂R. The reader should refer to [6] for details. Let us parameterize the curve as $\overrightarrow{C}(s,t) = [x(s,t) \;\; y(s,t)]$. The aim is to minimize the following energy function:

$$E = -\frac{1}{2}(u-v)^2 + \alpha \oint_{\overrightarrow{C}} ds, \quad (3)$$

where the parameters u and v denote the mean gray level intensities inside and outside the curve $\overrightarrow{C}$ and the second term is the length of the curve weighted by a constant α. Our aim is to move every point on the curve such that it moves in the negative direction of the energy gradient. After some manipulations (see [6]) the equation describing the motion of the curve is obtained as follows:

$$\frac{d\overrightarrow{C}(s,t)}{dt} = f(x,y)\overrightarrow{N} - \alpha\kappa\overrightarrow{N}, \quad (4)$$

$$f(x,y) = (u-v)\left(\frac{I(x,y)-u}{A_u} + \frac{I(x,y)-v}{A_v}\right), \quad (5)$$

which tells us to move each boundary point on the curve in a direction parallel to the normal vector drawn to the boundary at that point using a speed function derived from the image statistics and the curvature κ of the boundary defined at that boundary point. In the above equation $I(x,y)$ denotes a pixel intensity, and A_u, A_v denote areas inside and outside the curve. In [7], a polygonal implementation of the above curve evolution equation has been presented, which makes the implementation easier and faster, and has been adopted and generalized to pseudo-3D in this paper.

3.2. Pseudo-3D Generalization of Curve Evolution

Given a set of temporally unstable video object segmentation maps, we first stack them together so that a three-dimensional "object blob" in x-y-t space is formed (see Fig. 2 (a)). We propose to improve the temporal stability of this "object blob" by smoothing its surface using a surface evolution approach. If a polygonal surface is initialized so that it includes this "object blob", and if it is allowed to evolve so as to minimize its energy (3), it will eventually converge to a smoothed version of the 3D object volume. The smoothing effect is expected both due to the curvature term, which tries to make the surface as smooth as possible, and also due to the fact that the evolving surface is represented by polygonal patches which leaves out high curvature segments.

This 3D smoothing approach can be converted into a combination of simpler 2D smoothing steps by considering different cross sections (slices) of the "object blob" in x-y-t space. If we apply the curve evolution equation (4) to the segmentation maps in the x-y domain (at each t value), we can achieve spatial smoothness. In order to achieve temporal stability, we apply the curve evolution technique for each x-t and y-t cross section (slice) of the "object blob" iteratively as follows:

$$O^{n+1} = P_{yt}(P_{xt}(P_{xy}(O^n))) \quad (6)$$

where O^n denotes the "object blob" at iteration n and P_{yt} denotes the processing of each y-t cross-section of the "object blob" using a polygonal representation for $\overrightarrow{C}_{yt}(s,t)$:

$$\frac{\partial \mathbf{V}_{yt}(k)}{\partial t} = \widetilde{f}_{k,k-1}\overrightarrow{N}_{k,k-1} + \widetilde{f}_{k+1,k}\overrightarrow{N}_{k+1,k} - \alpha\kappa\overrightarrow{N}_b, \quad (7)$$

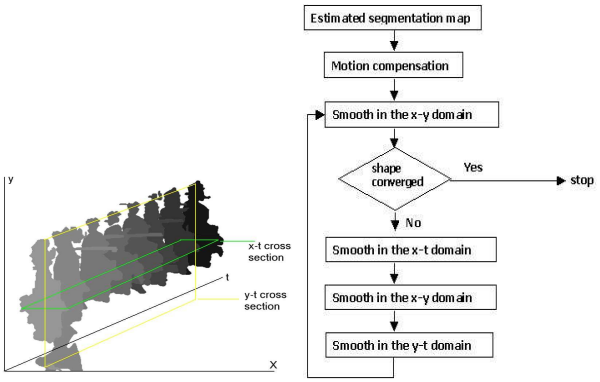
where $\mathbf{V}_{yt}(k)$ denotes a vertex on the polygonal boundary in the yt cross section of the "object blob", $\widetilde{f}_{k,k-1}$ and $\overrightarrow{N}_{k,k-1}$ denote the interpolated speed function (4) and the outward normal vector along the line connecting the vertices $\mathbf{V}(k)$ and $\mathbf{V}(k-1)$, respectively. The functions P_{xt} and P_{xy} are defined similarly.

This idea is illustrated in Fig. 2 (a), where the horizontal rectangle shows the x-t cross section and the vertical rectangle shows the y-t cross section. By using this pseudo-3D approach, we can obtain spatio-temporally stable object segmentation by processing the x-y, x-t and y-t slices of the "object volume" iteratively, until the shape convergences. The order of processing in the x-y, x-t and y-t slices does not produce significant changes in the experimental results.

Sometimes the y-t or x-t cross sections of the "object blob" do not consist of a single connected group of black pixels as can be observed in Fig. 5 (a), both due to the oscillatory (direction changing) motion of the object, and the natural topology of the object. The effect of the motion can be eliminated by motion compensating the binary object segmentation maps to align them with respect to the first frame. This transforms the 3D "object volume" into a more uniform block, thus minimizing the number of separate black regions in any y-t or x-t cross-section. If multiple disconnected black blobs still exist after motion compensation because of the natural topology of the object, the curve evolution has to be applied for each disconnected region of significant size. The overall flowchart of the proposed pseudo-3D smoothing algorithm is given in Fig. 2 (b).

4. EXPERIMENTAL RESULTS

The proposed pseudo-3D temporal stabilization algorithm is tested on the "Flikken" sequence (see Fig. 1), which is an extract from a



(a) (b)

Fig. 2. (a) The illustration of spatio-temporal smoothing. The gray-shaded regions stacked one after another represent the object segmentation maps in each frame. (b) The flowchart of the spatio-temporal smoothing using curve evolution.

TV movie. The segmentation of the objects in this realistic sequence is particularly difficult since the object and background colors are quite similar. The initial segmentation of the car, the walking lady and the man objects are carried out using the algorithm [8, 3]. The results on 168 frames of the "walking lady" object will be presented here. In Fig. 3, the smoothing results for several frames in the x-y domain are provided. The top row shows the given temporally unstable object segmentation maps and the bottom row shows the object segmentation maps after convergence of the curve. The weight of the curvature term in (3) is selected as $\alpha = 0.4$ (determined experimentally). We can observe that unwanted high-curvature parts and missegmented background regions are eliminated easily. In Fig. 4 (a), an x-t cross-section of the "lady" object for a fixed y value is shown (after processing in the x-y domain). The bottom figure shows the result of x-t curve evolution. We can see in Fig. 4 (a) that the elimination of the high curvature part in the x-t domain corresponds to the elimination of the missegmented background pixels in the x-y domain which is marked by the horizontal line in Fig. 4 (b). Fig. 4(b) shows the segmentation map of frame 111 in the spatial (x-y) domain before and after x-t processing. In Fig. 5 (a), a y-t cross section is given for a fixed x value. Two disconnected group of black regions can be seen due to the motion of the lady, who first walks towards left and then towards right. Motion compensation is utilized to make the cross sections more aligned, as seen in Fig. 5 (b). Fig. 5 (c) shows the y-t smoothing results. We can see that some high curvature lines are eliminated, which correspond to the legs of the lady, which actually introduces a loss of segmentation accuracy. However, this is not noticeable when the scene is viewed in 3D. The effect of y-t smoothing in the spatial domain is shown in Fig. 5 (d), where the temporal unstability caused by the legs is eliminated.

In Fig. 6, several frames of the Flikken sequence are shown after applying the complete spatio-temporal smoothing algorithm. We can see from the bottom row that the smoothed results do not display sudden changes as compared to the top row, which implies a better temporal stability. Although the accuracy of segmentation decreases in several frames after temporal stabilization, the overall decrease in segmentation accuracy for 168 frames was marginal (a 3% increase in the average number of missegmented pixels). However, this shows that it is possible to increase the quality of object segmentation without decreasing the segmentation errors, as explained below.

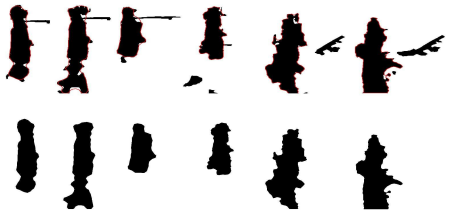
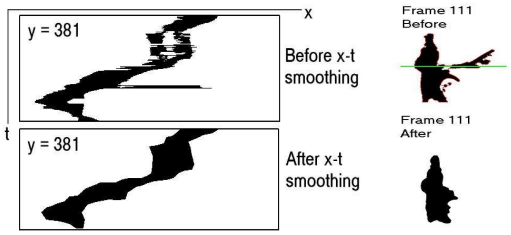


Fig. 3. **Processing in the x-y domain:** (Top Row) The original segmentation maps for frames 5, 9, 20, 85, 102, and 110. (Bottom Row) The results after processing in the x-y domain.



(a) (b)

Fig. 4. (a) **Processing in the x-t domain:** The x-t cross-section across 168 frames of the segmentation maps of the "lady" object before and after x-t smoothing. (b) Effects of x-t processing as observed in the x-y domain.

Objective Evaluation of the Results: In order to quantify the improvement in the temporal stability of the smoothed video object planes, they are evaluated using the histogram and shape measures, which were discussed in Section 2.

In Fig. 7(a), the plot of the histogram measure versus the frame number is given for the "lady" object, where large peaks at frame numbers such as $9, 10, 47, 48, \ldots$ correctly signal the frames where a large portion of the object has been removed from or added to the video object plane (see Fig. 1). Therefore, the histogram difference measure is a good indicator of the instants where we loose temporal stability.

In Fig. 7 (b), the plot of the histogram difference measure is given for the temporally stable video object planes, after processing with the proposed algorithm. If we compare the two plots (a) and (b), we can see that most of the peaks have been eliminated. Table 1 summarizes ratio of the mean and variance of the two plots, as well as the scores for the shape measure. We can observe that the mean and the variance of the histogram and shape measures are considerably smaller after spatio-temporal smoothing, indicating that the segmentation maps are more temporally stable.

Subjective (Perceptual) Evaluation of the Results: In order to see whether the proposed temporal stabilization algorithm improved the quality of 3D viewing in 3D-TV applications, we also carried out a set of perceptual evaluation tests. The depth information is added to a given 2D video sequence by segmenting the objects in the scene and then by placing each object at different inferred depths [3]. Then, the left and right views are rendered using a simple first-order extrapolation method for the disoccluded areas. Then the left and right sequences are displayed to the viewer using a set-up with glasses.

The objects in the Flikken sequence were also hand-segmented to obtain a reference (R) segmentation, with which the scenes ob-

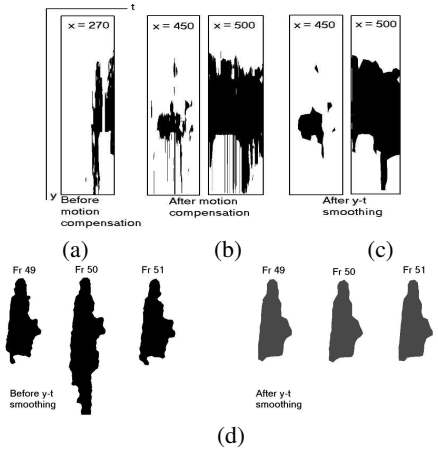


Fig. 5. **Processing in the y-t domain:** (a) A y-t cross-section of the "lady" object for a fixed x value. (b) Two y-t cross-sections after motion compensation. (c) The y-t cross-sections after y-t processing. (d) Effects of y-t domain smoothing as observed in the x-y domain for frames 49, 50 and 51.

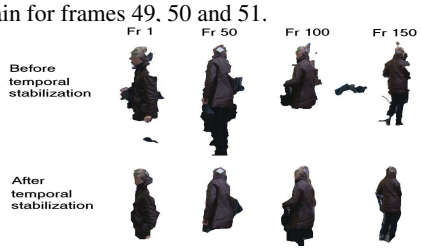


Fig. 6. **Top Row**: Original video object planes for frames 0, 50, 100 and 150. **Bottom Row**: The same frames after temporal stabilization.

tained by the unstable (U) and stable (S) object segmentation results are compared. During the perceptual tests, an observer was shown two stereo sequences A and B one after another. The sequences A and B can be one of the three cases R, U and S, giving us a total of nine combinations, named as Test1 - Test9. The observer was asked to select one of the choices: "B is significantly worse / slightly worse / the same as / slightly better / significantly better than sequence A. The five options are assigned the scores -2 to 2 from left to right, respectively.

The perceptual evaluation results for fourteen observers are summarized in Table 2. The tests where the two compared sequences A and B are exactly the same (such as UU, RR, SS) are used for checking the reliability of the tests, since they should have an average value of zero. The average score of the tests that compare S and U is 0.52, which indicates that S, the stabilized results are perceived as being better than the unstable results, when viewed in 3D. The average scores in Table 2 also indicate a quality ordering of the three cases as: $g(R) > g(S) > g(U)$, where $g(.)$ denotes the perceived quality of the rendered sequence.

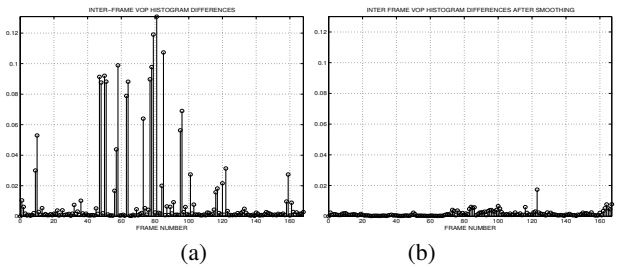


Fig. 7. The histogram difference measure between successive VOPs of the "lady" object before (a) and after (b) temporal stabilization versus frame number.

	Histogram Measure		Shape Measure	
	Mean	Var	Mean	Var
Before smoothing	11.52	696.76	38.87	158.69
After smoothing	1.64	3.83	9.90	36.22
$Ratio: \frac{Before}{After}$	7	182	3.9	4.4

Table 1. The ratio of the objective evaluation scores for the lady object before and after temporal stabilization, Histogram means and variances have been scaled by 10^3 and 10^6, respectively.

	Tests 1-2	Tests 3-4	Tests 5-7	Tests 8-9
AB pairs	-RU,UR	SR,-RS	UU,RR,SS	-SU,US
Av. Score	1.05	0.59	0.08	0.52

Table 2. Subjective evaluation scores for the Flikken sequence.

5. CONCLUSIONS AND FUTURE WORK

Obtaining temporally stable video object segmentation maps is important for comfortable viewing in 3D TV applications. In this paper, a pseudo-3D region-based curve evolution technique for temporally stabilizing a set of estimated video object planes has been introduced. It has been shown by experiments that the proposed algorithm significantly improves the temporal stability in terms of two quantitative objective measures based on histogram and shape differences. Subjective evaluation tests indicate that there is an improvement in the perceived quality of the scene when viewed in 3D, which also validates the effectiveness of the proposed quantitative measures. The experiments support our initial hypothesis that if there are inevitable object segmentation errors, they should be re-distributed in a temporally stable way. Hence, we conclude that it is possible to increase the object segmentation quality without increasing the segmentation accuracy. An object segmentation algorithm which optimizes the temporal stability measures directly is under development.

6. REFERENCES

[1] M. Op de Beeck and A. Redert, "Three dimensional video for the home," in *Proc. Int. Conf. On Augmented Virtual Environments and Three-Dimensional Imaging*, 2001, pp. 188–191.

[2] D. Zhang and G. Lu, "Segmentation of moving objects in image sequences: A review," *Circuits, Systems and Signal Processing*, vol. 20, no. 2, pp. 143–183, 2001.

[3] F. Ernst, "2d-to-3d video conversion based on time-consistent segmentation," in *Proc. ICOB'03 Workshop*, 2003.

[4] C. E. Erdem, B. Sankur, and A. M. Tekalp, "Performance measures for video object segmentation and tracking," *IEEE Transactions on Image Processing*, vol. 13, no. 7, 2004.

[5] E. M. Arkin, L. P. Chew, D. P. Huttenlocker, K. Kedem, and J. S. B. Mitchell, "An efficient computable metric for comparing polygonal shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, pp. 209–215, 1991.

[6] A.Yezzi, A. Tsai, and A. Willsky, "A fully global approach to image segmentation via coupled curve evolution equations," *Journal of Visual Communication and Image Representation*, vol. 13, pp. 195–216, 2002.

[7] G. Unal, H. Krim, and A.Yezzi, "A vertex-based representation of objects in an image," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, 2002, vol. 1, pp. 896–899.

[8] F. Ernst, P. Wilinski, and K. van Overveld, "Dense structure-from-motion: An approach based on segment matching," in *Proceedings of European Conference on Computer Vision*, 2002.