Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Signal Processing: *Image Communication* 20 (2005) 151–167

SIGNAL PROCESSING: IMAGE COMMUNICATION

www.elsevier.com/locate/image

Temporal stabilization of Video Object Segmentation for 3D-TV applications

Çiğdem Eroğlu Erdem[a,*], Fabian Ernst[b], Andre Redert[b], Emile Hendriks[c]

[a] *Research Department, Momentum Digital Media Technologies A.Ş. TÜBİTAK - MAM - TEKSEB, A - 205, Gebze, Kocaeli, Turkey*
[b] *Philips Research Laboratories, Prof. Holstlaan 4, 5656 AA, Eindhoven, The Netherlands*
[c] *Department of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Information and Communication Theory Group, Mekelweg 4, 2628 CD, Delft, The Netherlands*

Received 29 February 2004

Abstract

Our aim is to insert depth information into an existing 2D video sequence to provide content for 3D-TV applications, which we try to achieve through segmentation of the objects in the given 2D video sequence. To this effect, we present a method for temporal stabilization of video object segmentation algorithms for 3D-TV applications. First, two quantitative measures to evaluate temporal stability without ground-truth are discussed. Then, a pseudo-3D curve evolution method, which spatio-temporally stabilizes the estimated segmentation of a video object 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 is inevitable, these errors 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. Therefore, it is possible to increase the perceptual object segmentation quality without increasing the actual segmentation accuracy.

Keywords: Object segmentation; Object tracking; Temporal stabilization; 3D TV; Curve evolution; Snakes

1. Introduction

The task of building three dimension (3D) models of a time-varying scene, using the 2D image sequence recorded by a single camera is an important but unsolved task to provide content for 3D-TV applications [20]. Providing content for the

*Corresponding author. Tel.: +90 262 641 6126; fax: +90 262 641 6137.

E-mail addresses: cigdem@ieee.org (Ç.E. Erdem), fabian.ernst@philips.com (F. Ernst), andre.redert@philips.com (A. Redert), E.A.Hendriks@ewi.tudelft.nl (E. Hendriks).

0923-5965/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.image.2004.10.005

newly emerging 3D-TV will be an important task. Although there are some other techniques which use two or more cameras [16], depth sensors or structured light [22] to extract the depth information of a scene more accurately, these methods require extra resources and may only be used for recording new video material. Therefore, it is desirable and important to convert the vast amount of already existing 2D video material into 3D. Given the restriction that we have a single 2D video sequence recorded by a single camera, it is quite challenging to form a complete 3D model of the scene. In this paper, a relatively simple approach is followed for 3D-TV applications. This approach consists of segmenting the objects in a given 2D video and ordering their video object planes (VOPs) with respect to their relative depths. Then, a left and a right view of the scene are rendered to obtain a stereo sequence, which yields a satisfactory sense of 3D. The above approach is illustrated in Fig. 1, where the three objects in the scene (lady, man and the car) are segmented and ordered for rendering the left and the right views. The rendering of the left and right views requires some extrapolation for the uncovered pixels. The intra-object depth variations can also not be displayed with this depth layering approach. However, the method gives satisfactory results when the objects are sufficiently far from the camera, so that the intra-object depth variations can be ignored.

One of the most important requirements is the *temporal stability* of the video object planes. Ideally, VOPs should follow the shape and color profile of the real 3D object. 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, 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. Some automatic object segmentation methods assume that the background is stationary and the object of interest is moving independently of background motion. Other methods impose constraints on the shape of the tracked object by using 2D [7,18] or 3D shape [6,25] models. Some of these algorithms acquire the 2D shape space information through training [5,3,17], while others require an initial [11,27] or global object model which may be provided by the user. For 3D-TV applications an unsupervised and automatic object segmentation method is needed. Motion segmentation-based approaches fall into this category [14,13]. After a watershed-based color segmentation of each frame, the color segments are grouped into object based on their motion. 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 (see Fig. 2(c)). In this paper, we try to

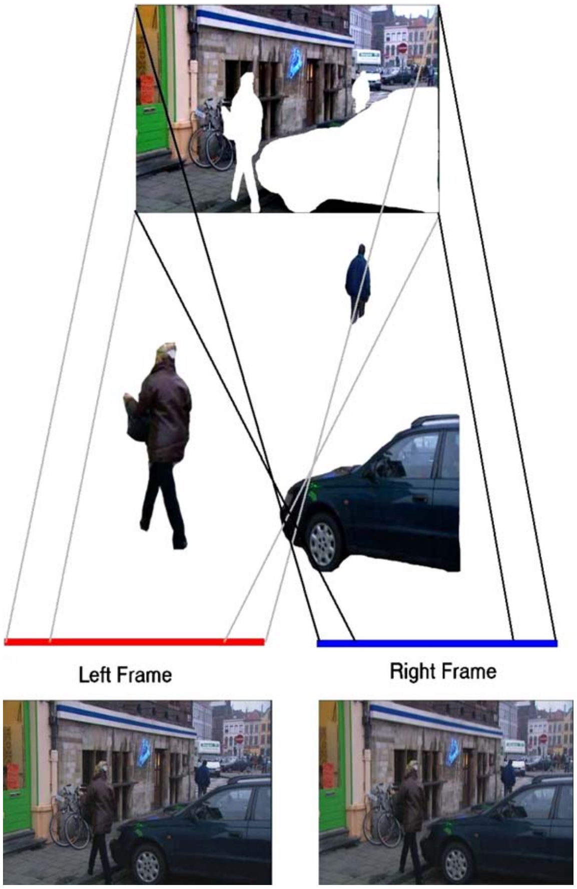


Fig. 1. Depth insertion to a 2D video using object segmentation.

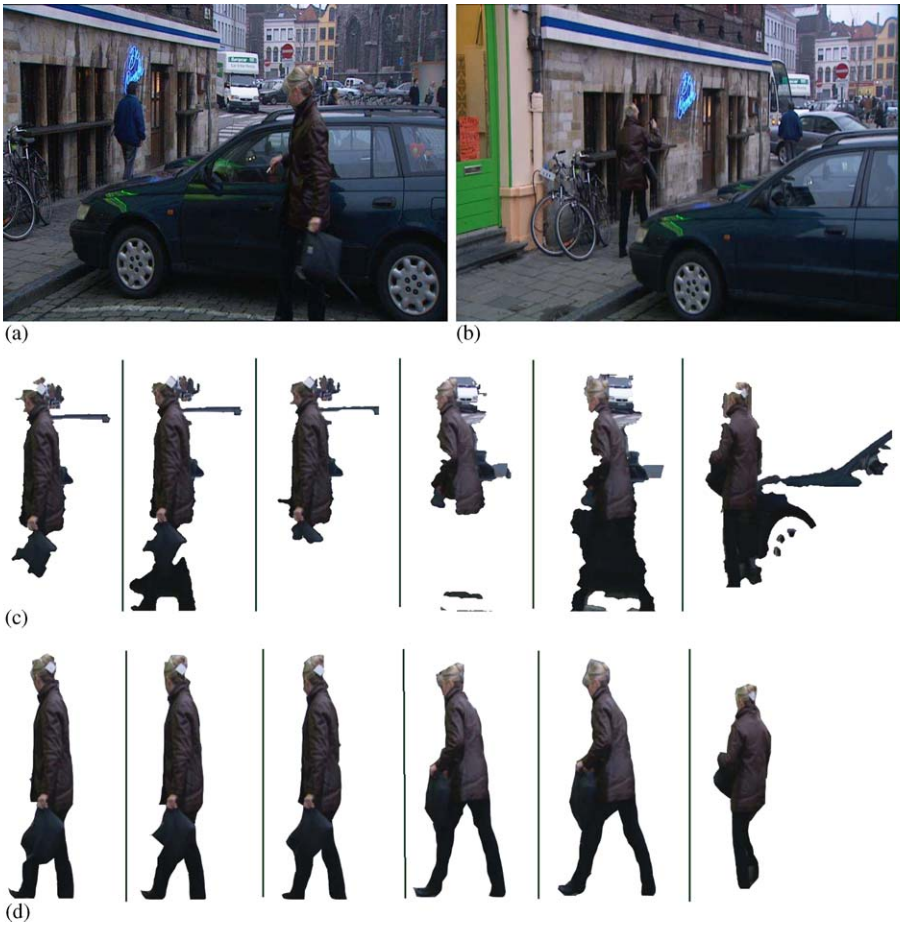


Fig. 2. (a), (b) First and last frames of "Flikken" sequence. (c) The given temporally unstable video object planes for the "lady" object (frames 8, 9, 10, 80, 81, 112) from left to right. (d) Ground-truth VOPs for frames 8, 9, 10, 80, 81, 112.

answer the question: "If object segmentation errors are inevitable, how can we conceal them in our application?". Our approach is based on the hypothesis that if segmentation errors are inevitable, they should be temporally consistent to increase the viewing comfort in 3D-TV applications. To this effect, we propose a pseudo-3D curve evolution technique, which re-distributes the existing segmentation errors such that they will be less visible when the scene is rendered and viewed in stereo.

Object tracking methods which use spatio-temporal surface evolution have been recently proposed [15,19]. Since they utilize level sets which are based on evolving a surface, they are computationally more complex as compared to the pseudo-3D algorithm presented in this paper, which uses polygon approximations to the object geometry in 2D cross-sections of the 3D object volume.

There are a few performance measures in the literature to evaluate the quality of video object segmentation maps without ground-truth [10,12,8]. In [8], the temporal stability is quantified in terms of object area, shape (elongation) and texture differences between two successive frames. However, this approach only monitors and compares the temporal stability of different segmentation results.

The contribution of this paper is twofold. First, two measures to quantitatively evaluate the "temporal stability" of a given set of video object planes will be discussed, which are based on the inter-frame color histogram and shape differences of the video object planes. Then, a novel post-processing algorithm to improve the temporal stability of a given set of video object planes will be introduced, which is based on pseudo-3D curve evolution. The input to the proposed algorithm is a set of temporally unstable object segmentation maps which can be estimated by any algorithm in the literature. The results provided by the algorithm [14,13] will be used in this paper.

The organization of this paper is as follows. In Section 2, the temporal stability measures are discussed. The details of the proposed temporal stabilization method are provided in Section 3 and the experimental results are given in Section 4. Finally, conclusions are provided in Section 5.

2. Measures for temporal stability

In this section, we discuss two quantitative measures for evaluating temporal stability of video object segmentation. The two measures use the histogram and shape differences between two successive VOPs, respectively. 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 VOPs. Estimated VOPs lack "temporal stability" mainly in two different ways: either large object regions are excluded or large background regions are included in the estimated maps by mistake (see Fig. 2(c)), causing large differences between histograms of successive VOPs. One shortcoming of the histogram measure is that it cannot detect if a portion of the object has been 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 two successive video object planes are two measures for indicating the temporal stability of the object segmentation.

2.1. Histogram differences between video object planes

There are various ways to find the difference between two histograms [12,10]. The chi-square based measure has been reported [10] to be more distinctive as compared to other techniques. The difference between two histograms can be calculated using the chi-square measure as follows [12]:

$$d_{\chi^2}(H_{t-1}, H_t) = \frac{\sum_{j=1}^{B} \frac{[r_1 H_{t-1}(j) - r_2 H_t(j)]^2}{H_{t-1}(j) + H_t(j)}}{N_{H_{t-1}} + N_{H_t}}, \tag{1}$$

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, and normalization parameters are defined as follows:

$$r_1 = \sqrt{\frac{N_{H_{t-1}}}{N_{H_t}}}, \quad r_2 = \frac{1}{r_1}, \quad N_{H_t} = \sum_{j=1}^{B} H_t(j).$$

The single-object definition above can be extended to multiple objects to get a single measure for the whole frame. One way of doing this is to compute (1) for each object and take the maximum or average over all objects in the frame [18].

2.2. Shape differences between video object planes

One way to represent the "shape" of a video object is to use the turning angle function of the boundary pixels [1]. In [21], a comparison of shape difference calculation methods has been reported, which concludes that the most successful method of shape representation (for image retrieval) is based on the turning angles.

The turning angle function (TAF) plots the counter clockwise angle from the x-axis as a function of the boundary length [1]. For details see Ref. [1]. After obtaining the TAFs belonging to the video objects in successive frames, which are 1D functions describing the shapes (denoted by θ_t

and θ_{t-1}), the distance between them is calculated as follows:

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

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. Before calculating difference (2), the number of points in both TAFs are equalized by resampling them. Definition (2) for one object can be extended to include multiple objects as discussed in the previous section.

3. Temporal stabilization of video object segmentation maps

In this first part of this section, an overview of the curve-evolution technique that we will adopt is given. Region-based curve evolution techniques have been used for image segmentation in the literature [27,26,24,2,4]. The main assumption behind these techniques is that the region to be segmented can be characterized by a predetermined set of features such as mean, variance, and texture, which may be inferred from the image data. For details of this method, the reader is referred to [26].

In the second part of this section, we will present a pseudo-3D curve evolution method, which spatio-temporally stabilizes a given set of video object planes. The curve evolution technique is a suitable tool to achieve this, since the "object volume" (consisting of VOPs over time) is characterized by a constant gray-level value. That is, we represent the video object planes, (or any other cross-section of the "object volume") as a binary image, where the relevant object region is black and the remainder of the image is white. This is the image on which the curve evolution technique is applied, as will be clarified below.

3.1. Background theory

A simple image segmentation problem is the case where there are just two types of regions in the image as shown in Fig. 3(a). We start with an arbitrary initialization of the segmentation boundary as denoted by $\vec{C}$. Then, the curve $\vec{C}$ will be evolved in such a way that it will eventually snap to the desired object boundary ∂R, which is the boundary of the dark region in Fig. 3 [26]. Let us parameterize the curve as $\vec{C}(s,\tau) = [x(s,\tau)\ y(s,\tau)]$, where s denotes the arc length of the curve and τ denotes the iteration number corresponding to the evolution of the curve in a certain image. The aim is to minimize the following energy function:

$$E = -\frac{1}{2}(u - v)^2 + \alpha \oint_{\vec{C}} \mathrm{d}s, \tag{3}$$

where the parameters u and v denote the mean gray level intensities inside and outside the curve $\vec{C}$ 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 $\vec{C}$ such that it moves in the negative gradient direction of the energy. The equation describing the motion of the curve is obtained as follows:

$$\frac{\mathrm{d}\vec{C}(s,\tau)}{\mathrm{d}\tau} = -\nabla E. \tag{4}$$

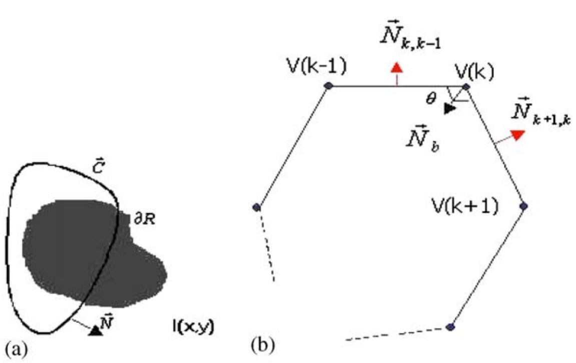


Fig. 3. (a) The aim is to segment the dark region in the image $I(x,y)$. The black curve C is initialized arbitrarily. (b) The polygon representation of the curve.

After some manipulations (see [26] for details) we obtain:

$$\frac{\mathrm{d}\vec{C}(s,\tau)}{\mathrm{d}\tau} = f(x,y)\vec{N} - \alpha\kappa\vec{N}, \tag{5}$$

$$f(x,y) = (u-v)\left(\frac{I(x,y)-u}{A_u} + \frac{I(x,y)-v}{A_v}\right), \tag{6}$$

which tells us to move each point on the curve in a direction parallel to the normal vector at that point using a speed function derived from the image statistics and the curvature of the boundary κ 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. The curvature term prevents the curve to be effected from noise, but on the other hand sharp corners are rounded off.

In [24], a polygonal implementation of curve evolution has been proposed, which makes the implementation easier and faster. This approach will be adopted and generalized to pseudo-3D as described in the next section. In [24], the curve is represented using a fixed number of vertices $\{V(0),\ldots,V(n)\} = \{(x_i,y_i), i=1,\ldots,n\}$ (see Fig. 3(b)). Then, the curve $\vec{C}$ is parametrized by a parameter $p \in [0,n]$ as follows:

$$\vec{C}(p,V) = L(p-\lfloor p\rfloor, V(\lfloor p\rfloor), V(\lfloor p\rfloor+1)). \tag{7}$$

In the above equation, $\lfloor p\rfloor$ denotes the largest integer not greater than p, and $L(\tau,X,Y) = (1-\tau)X + \tau Y$ is linear interpolation of vertices X and Y where τ changes between 0 and 1. If the vertex-based notation is used in (4), then the energy is minimized by solving a set of ODEs for each vertex (see [24] for derivation). The evolution of the curve towards the minimum energy location can now be described through the motion of its vertices:

$$\begin{aligned}\frac{\partial V(k)}{\partial\tau} = &\int_0^1 pf(L(p,V(k-1),V(k)))\,\mathrm{d}p\vec{N}_{k,k-1}\\ &+\int_0^1 pf(L(p,V(k),V(k+1)))\,\mathrm{d}p\vec{N}_{k+1,k}\\ &-\alpha\kappa\vec{N}_b,\end{aligned} \tag{8}$$

where $\vec{N}_{k,k-1}$ denotes the normal vector of the polygon edge connecting the vertices $V(k-1)$ and $V(k)$, as shown in Fig. 3(b). The utilization of polygons decreases the amount of computational and memory costs.

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 3D "object blob" in $x-y-t$ space is formed (see Fig. 4). 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 processing 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

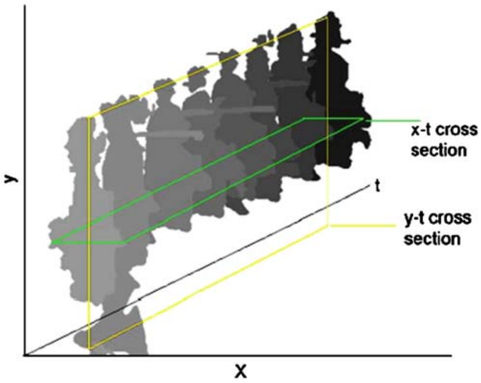


Fig. 4. The illustration of spatio-temporal stabilization. The object segmentation maps in successive frames are shown by gray-shaded regions.

(slice) of the "object blob" iteratively as follows:

$$O^{n+1} = \Im_{yt}(\Im_{xt}(\Im_{xy}(O^n))), \tag{9}$$

where O^n denotes the "object blob" after iteration n. The symbols $\Im_{yt}$, $\Im_{xt}$ and $\Im_{xy}$ denote the processing of each $y-t$, $x-t$ and $x-y$ cross-section of the "object blob" using a polygonal representation for the boundary of that cross-section (e.g. for $\vec{C}_{yt}(s,t)$) as follows:

$$\frac{\partial V_{yt}(k)}{\partial \tau} = \tilde{f}_{k,k-1}\vec{N}_{k,k-1} + \tilde{f}_{k+1,k}\vec{N}_{k+1,k} - \alpha\kappa\vec{N}_b, \tag{10}$$

where $V_{yt}(k)$ denotes a vertex on the polygonal boundary in a yt cross section of the "object blob" (see Fig. 3(b)), $\tilde{f}_{k,k-1}$ and $\vec{N}_{k,k-1}$ denote the interpolated speed function and the outward normal vector of the line connecting the vertices $V(k)$ and $V(k-1)$, respectively. The interpolated speed function is defined as follows:

$$\tilde{f}_{k,k-1} = \int_0^1 pf((1-p)V(k-1) + pV(k))\,\mathrm{d}p. \tag{11}$$

The processes $\Im_{xt}$ and $\Im_{xy}$ for $x-t$ and $x-y$ cross-sections have definitions similar to (10).

This idea is illustrated in Fig. 4, where the horizontal rectangle shows an $x-t$ cross section and the vertical rectangle shows a $y-t$ cross section of the "object blob". Using the above pseudo-3D curve evolution approach, spatio-temporal stability can be achieved by processing the $x-y$, $x-t$ and $y-t$ slices of the "object volume" iteratively, until the shape converges.

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. 8(a), both due to the oscillatory (direction changing) motion of the object, and also 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.

In order to carry out the motion compensation step, we first estimate the motion parameter of each object at each frame. The motion vectors of the color segments estimated during object segmentation [14,13] is used for this purpose. In [14,13], first a watershed-based color segmentation of each frame is carried out and then, a motion vector is estimated for each color segment. Finally, the color segments are grouped into objects using their estimated motion vectors. These estimated motion vectors were used to align the video object planes of successive frames with respect to the first frame, so that both the effects of camera motion and object motion are minimized. The more accurate the motion compensation is, the better the alignment will be. However, utilization of a simple motion estimation paradigm was found to be sufficient during the experiments.

If multiple disconnected black blobs still exist after motion compensation because of the natural topology of the object, the curve evolution is applied to each disconnected region of significant size separately. The overall flowchart of the proposed pseudo-3D smoothing algorithm is given in Fig. 5. Since all the segmentation maps need to be processed all at once, the algorithm runs off-line.

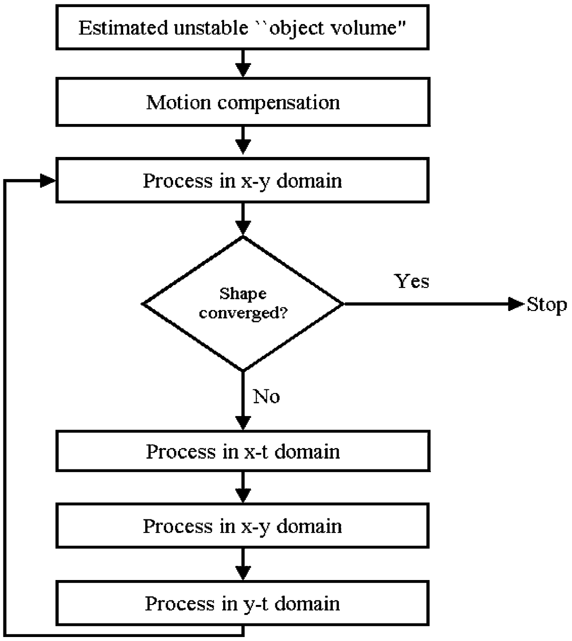


Fig. 5. The flowchart of the spatio-temporal smoothing using curve evolution.

4. Experimental results

In this section, we first provide the experimental results on the "Flikken" sequence to show that spatio-temporal smoothing of the object segmentation maps using curve evolution improves the temporal stability in terms of the two measures discussed in Section 2. We use the video object planes estimated by algorithm [14,13] as input. This algorithm has three main processing steps. In the first step, the spatial color segmentation of each frame is estimated using a watershed-based region-growing algorithm. It is assumed that the boundaries of the actual objects in the scene are a subset of the boundaries of the estimated color segments. In the second step, the translational motion vector of each color segment is estimated using a region-matching procedure. Finally in the last step, these motion vectors (and hence the color segments) are grouped to yield the object boundaries. Sample frames of the original sequence and the estimated temporally unstable VOPs are given in Fig. 2. Especially notice the large changes between neighboring frames 8, 9, 10 and 80, 81, which cause temporal unstabilities. The Flikken sequence is a difficult scene, especially for the segmentation of the lady object. This is because the color differences between clothes of the lady and the car are much smaller than the color differences within the car (e.g. sides and the windows). The lack of texture also makes motion estimation difficult. Therefore, most color segmentation-based (or block based) motion segmentation algorithms are expected to yield unstable results in this scene.

As was discussed in the introduction, our final aim is to convert 2D video sequences to 3D (stereo) so that they will be perceptually pleasing. In order to assess whether the proposed pseudo-3D curve evolution approach improves the comfort of 3D viewing, perceptual evaluation tests have been done. The procedure followed in the perceptual tests and the results will be provided in the second part of this section.

4.1. Spatio-temporal smoothing of the Flikken sequence

The proposed pseudo-3D smoothing algorithm is tested using frames 1–168 in shot 5 of the "Flikken" sequence,[1] and the results on the "lady" and the "man" objects will be presented.

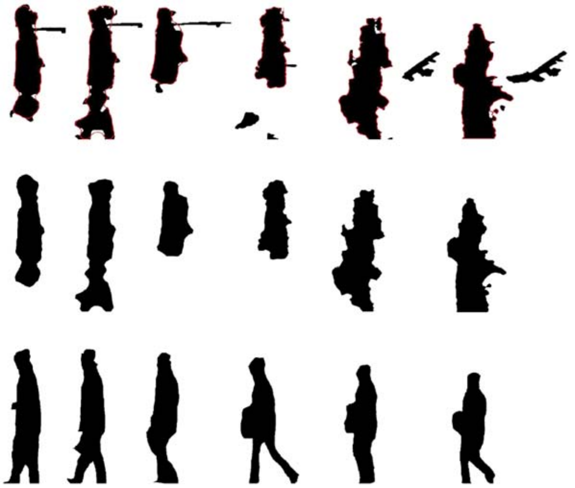


Fig. 6. *Processing in the $x-y$ domain*: Top row shows the original segmentation maps for frames 5, 9, 20, 85, 102, and 110 from left to right. Middle row shows the segmentation maps after $x-y$ domain processing. Bottom row shows the ground-truth segmentation maps for the same frames.

4.1.1. Lady object

In Fig. 6, the results for several frames in the $x-y$ domain are provided. The top row shows the given object segmentation maps and the bottom row shows the smoothed object segmentation maps after convergence of the curve. The weight of the curvature term is selected as $\alpha = 0.4$, which is determined experimentally. We can observe that unwanted high-curvature parts and missegmented background regions have been eliminated easily.

In the top part of Fig. 7(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 after processing in the $x-t$ domain. Fig. 7(b) shows the segmentation map of frame 111 in the spatial $(x-y)$ domain before and after $x-t$ domain processing. We can see that, the elimination of the high curvature part in the $x-t$ domain (see Fig. 7(a)) corresponds to the elimination of the missegmented background pixels in the $x-y$ domain, which

[1] A TV series produced by MMGNV for VRT.

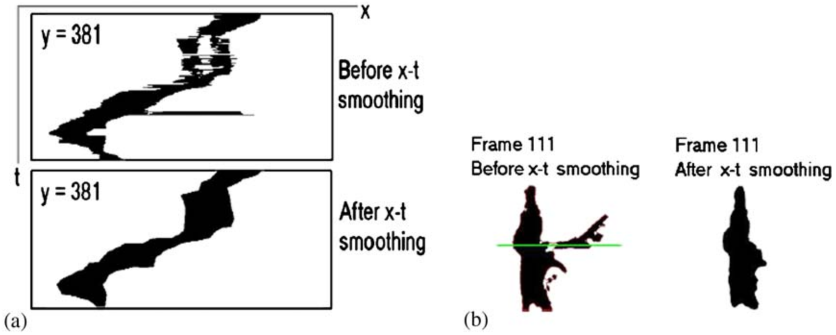


Fig. 7. *Processing in the* $x-t$ *domain*: The $x-t$ cross-section across 168 frames of the segmentation maps of the "lady" object before and after $x-t$ processing. (b) Effects of $x-t$ processing as observed in the $x-y$ domain. *Left*: Black pixels denote the original segmentation map. The horizontal line corresponds to the $x-t$ high-curvature regions in (a). *Right*: The segmentation map after $x-y$ and $x-t$ smoothing.

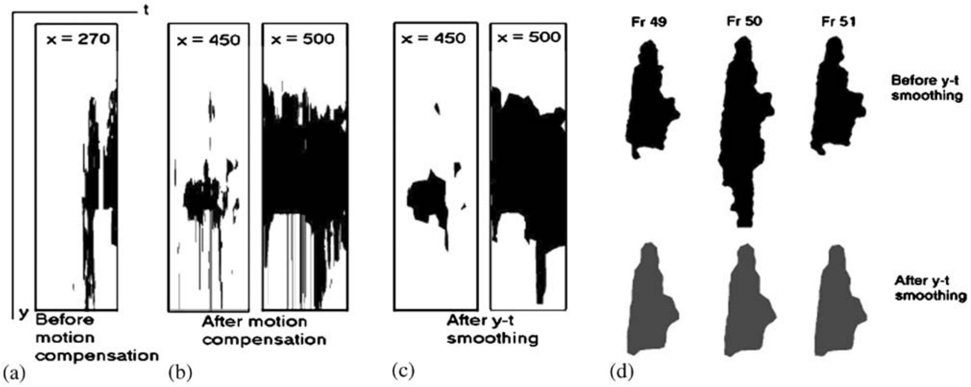


Fig. 8. *Processing in the* $y-t$ *domain*: (a) A $y-t$ cross-sections of the "lady" object for a fixed x value. Note that due to the oscillatory motion of the object in the x direction, we have two disconnected black blobs. (b) Two $y-t$ cross-sections after motion compensation. (c) The $y-t$ cross-sections after $y-t$ processing. (d) Effects of $y-t$ domain processing as observed in the $x-y$ domain for frames 49–51 (from left to right). The disturbing temporal unstability is eliminated.

is marked by the horizontal line in Fig. 7(b). In Fig. 8(a), a $y-t$ cross section of the "object blob" for the lady object is given for a fixed x value. Two disconnected group of black regions can be seen due to the oscillatory motion of the object (walking towards left and then towards right). We used motion compensation to make the "object blob" more aligned in time. In Fig. 8(b), two $y-t$ cross sections of the "lady" object after motion compensation are given, which show a better alignment (no disconnected regions). Fig. 8(c) shows the results after $y-t$ domain processing. We can see that some high-curvature lines are eliminated, which actually correspond to the legs of the lady. The effects of $y-t$ processing as observed in the spatial domain are given in Fig. 8(d), which shows that the temporal unstability caused by the legs is eliminated.

In Fig. 9, several frames of the Flikken sequence are shown before and after applying the complete temporal stabilization algorithm together with the ground-truth video object planes. We can see from the middle row that the processed VOPs do not display sudden changes as compared to the top row, which implies a better temporal stability. The accuracy of object segmentation (i.e. the number of misclassified pixels) decreases in several frames after temporal stabilization with respect to the

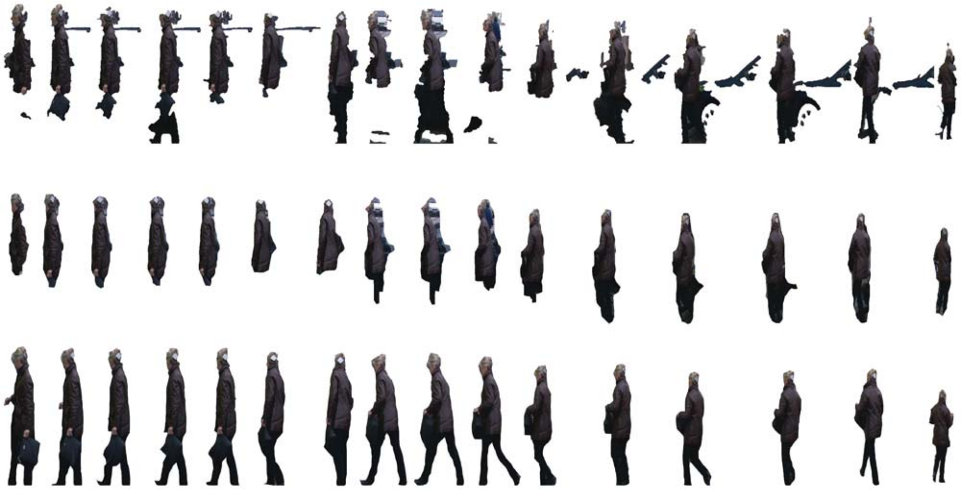


Fig. 9. Temporal stabilization results for the lady object. *Top row*: Given unstable video object planes for frames 1, 5, 8, 9, 10, 20, 50, 80, 81, 85, 100, 102, 110, 112, 116, and 150 from left to right. *Middle row*: The same frames after temporal stabilization. *Bottom row*: The ground-truth segmentation maps for the same frames.

unstable results. For example see Fig. 9, fourth column. On the other hand the accuracy increases in other frames, as shown in Fig. 9, final columns. The average decrease in segmentation accuracy averaged over 168 frames was marginal (a 3% increase in the number of missegmented pixels with respect to the given unstable results). This shows that it is indeed possible to increase the quality of object segmentation without decreasing the spatial segmentation errors, as will be in the next section.

In Fig. 10, the ratio of misclassified pixels with respect to the ground-truth segmentation is plotted for every frame. We can see that the given unstable segmentation maps have about 43% pixel misclassification error on the average. This number increases to 45% after temporal stabilization. Although there is a slight increase in the actual number of misclassified pixels, the errors are distributed more uniformly after temporal stabilization, which provides a better quality in our application of 3D TV.

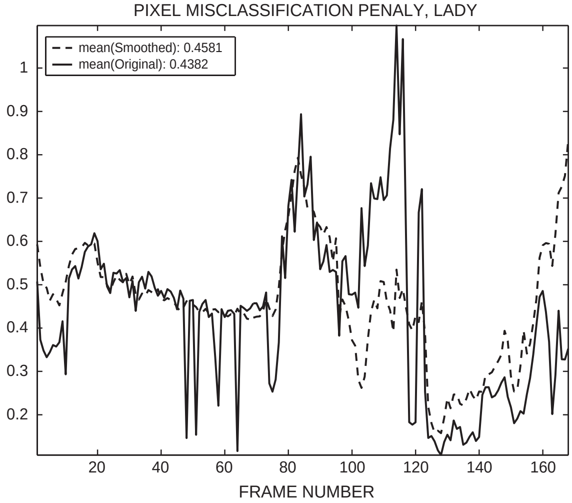


Fig. 10. The ratio of misclassified pixels with respect to the ground-truth segmentation for every frame, for the lady object. The solid line denotes the original unstable results, and the dashed line denotes the results after temporal stabilization.

4.1.2. Man object

The stabilization results for the man object are given in Fig. 11, together with the ground-truth segmentation maps. We can observe that the stabilization algorithm eliminates the noisy parts and makes the overall object contours smoother and hence less disturbing for the eye. Fig. 12 plots the ratio of misclassified pixels with respect to the ground-truth segmentation for every frame, which indicates a slight increase in the number of misclassified pixels after temporal stabilization.

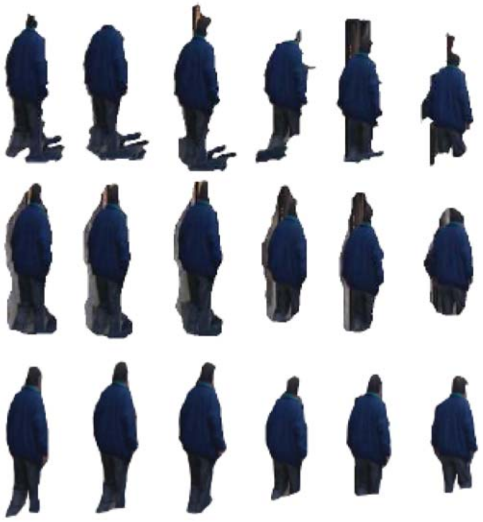


Fig. 11. Temporal stabilization results for the man object. *Top row*: Given unstable video object planes for frames 19, 20, 21, 44, 46, and 61 from left to right. *Middle row*: The same frames after temporal stabilization. *Bottom row*: The ground-truth segmentation maps for the same frames.

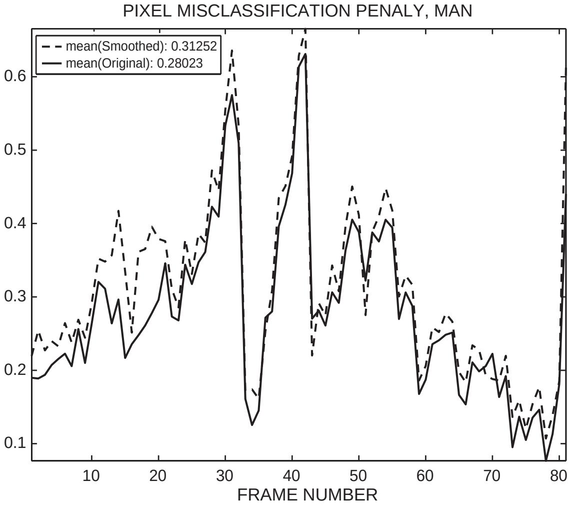


Fig. 12. The ratio of misclassified pixels with respect to the ground-truth segmentation for every frame, for the man object. The solid line denotes the original unstable results, and the dashed line denotes the results after temporal stabilization.

4.2. Objective evaluation of the results

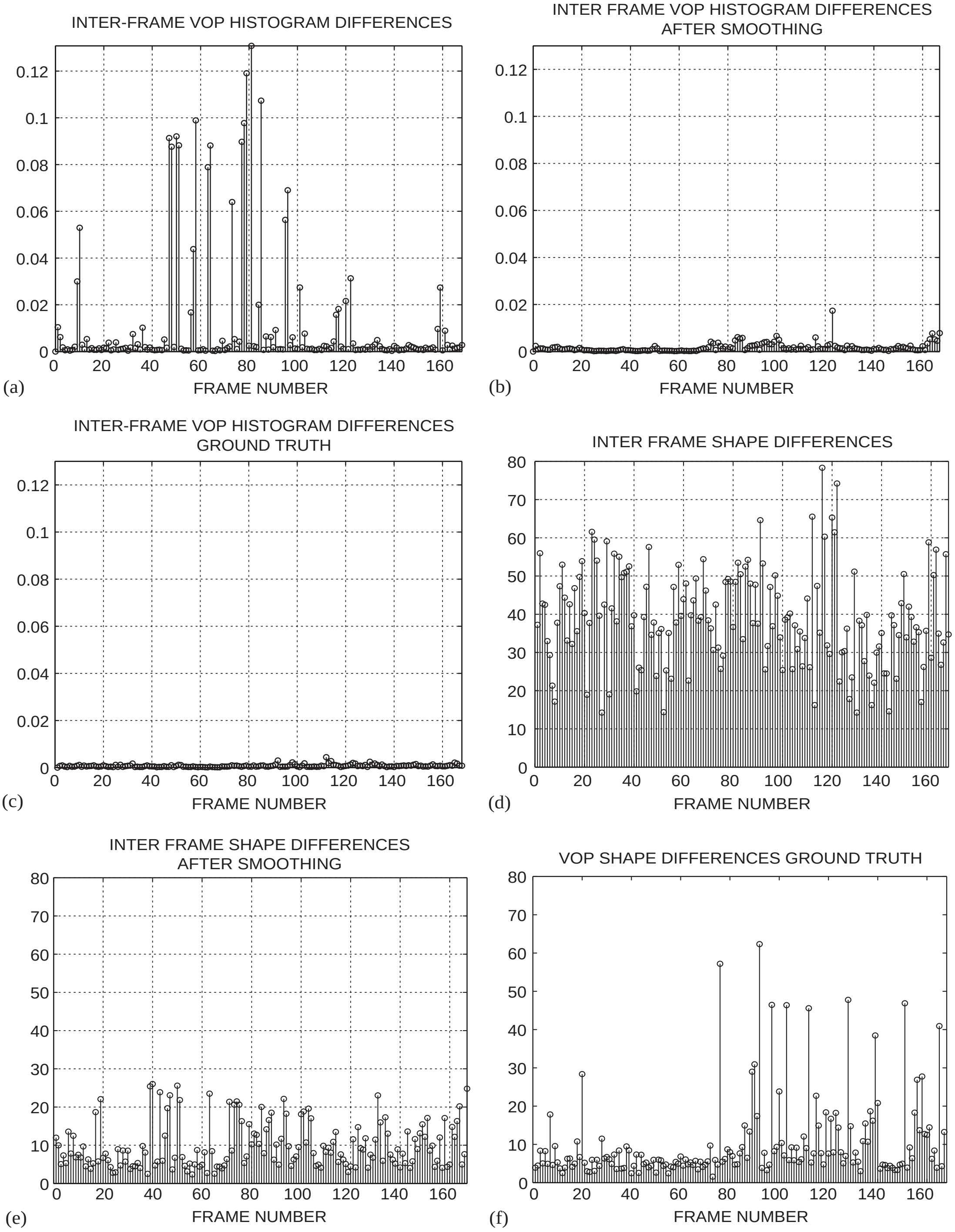
In order to quantify the improvement in the temporal stability of the smoothed video object planes, they are evaluated using the histogram and shape measures, which were discussed in Section 2.

4.2.1. Lady object

In Fig. 13(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, 80, 81 ... can be seen. If we look at the actual video object planes, which are given in Fig. 2(c), we can see that the histogram measure correctly signals the frames where a large portion of the object has been removed from or added to the video object plane. Therefore, the histogram difference measure is a good indicator of the instants where we loose temporal stability. However, it does not tell us which segmentation is "good" or "bad", it only signals a "big change" in the object segmentation.

In Fig. 13(b), the plot of the histogram difference measure is given for the spatio-temporally smoothed video object planes. If we compare the two plots (a) and (b), we can see that most of the peaks have been eliminated. In Fig. 13(c) the histogram measure for the ground-truth segmentation maps is given. Table 1 summarizes the mean and variance of the two plots and their ratio. We can observe that the mean and the variance of the histogram measure are considerably smaller after spatio-temporal smoothing, indicating that the segmentation maps are more temporally stable.

In Fig. 13(d) and (e), the plot of the shape measure before and after spatio-temporal smoothing is given. The mean and variance values are summarized in Table 1, which shows that both the mean and the variance decreased considerably after spatio-temporal smoothing. However, the amount of decrease is also important should be interpreted more carefully. Fig. 13(f) gives the shape measure for the ground-truth segmentation maps, which shows some large peaks which actually correspond to the large shape differences arising from the crossing legs when the lady walks. Since the segmentation of the legs were very problematic in the given unstable segmentation maps, we observe that they are over-smoothed after temporal stabilization.


INTER-FRAME VOP HISTOGRAM DIFFERENCES

INTER FRAME VOP HISTOGRAM DIFFERENCES AFTER SMOOTHING

INTER-FRAME VOP HISTOGRAM DIFFERENCES GROUND TRUTH

INTER FRAME SHAPE DIFFERENCES

INTER FRAME SHAPE DIFFERENCES AFTER SMOOTHING

VOP SHAPE DIFFERENCES GROUND TRUTH

Fig. 13. Lady Object: (a), (b): The histogram difference measure before (a) and after (b) spatio-temporal stabilization versus frame number. (d), (e): The shape difference measure before (d) and after (e) spatio-temporal stabilization.

Table 1
Objective evaluation scores for the lady object before and after spatio-temporal smoothing

	Histogram measure		Shape measure	
	Mean ($\times 10^{-3}$)	Variance ($\times 10^{-6}$)	Mean	Variance
Before smoothing	11.52	696.76	38.87	158.69
After smoothing	1.64	3.83	9.90	36.22
Ratio (before/after)	7	182	3.9	4.4
Ground-truth	0.52	0.09	10.20	115.55

4.2.2. Man object

In Fig. 14, the histogram and shape measures are plotted for the man object at every frame before and after temporal stabilization together with the plots for the ground-truth segmentation maps. The frame number goes up to 80 for the man object since the object leaves the scene after that. The mean values of these plots are summarized in Table 2. We can see that there is a decrease in the histogram and shape difference measures, which is desirable.

4.3. Subjective (perceptual) evaluation of the results

As was discussed in the introduction section, our final aim is to create a sense of 3D in a given 2D video sequence. Currently, the depth information is added to a 2D video sequence by segmenting the objects in the scene and then by placing them at different depths. (The relative ordering of the objects in the scene can be inferred using occlusion information [13,23].) Then, left and right sequences are rendered, which are finally viewed in a stereo set-up with glasses.

In order to assess whether the proposed spatio-temporal smoothing scheme improves the comfort level of 3D viewing, we carried out a set of perceptual evaluation tests. The Flikken sequence was segmented into its objects by hand to obtain an almost perfect segmentation, which will be used as a benchmark in the perceptual tests. The goal of the perceptual tests presented in this section is to assess how the rendered stereo sequences obtained from the hand segmented (H), the unstable (U) and the stable (S) object segmentation results are perceived and ranked by a human observer. During the perceptual tests, an observer was shown two stereo sequences A and B, one after the other under controlled viewing conditions [9]. The sequences A and B can be one of the three cases H, U and S, giving us a total of nine combinations, named as Test 1–Test 9. The observer was asked to answer the question: "How do you compare sequence B to sequence A?", the answer of which is given as choosing one of the options "B is significantly worse/slightly worse/the same as/slightly better/significantly better than sequence A." The five options are assigned the scores $-2, -1, 0, 1$ and 2 from left to right, respectively.

The perceptual evaluation results for fourteen observers are summarized in Table 3. The tests where the two compared sequences A and B are exactly the same (such as UU, HH, 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 temporally stabilized results are perceived as being better than the unstable results, when viewed in 3D. The average scores in Table 3 also indicate a quality ordering of the three cases as: $g(\mathrm{H}) > g(\mathrm{S}) > g(\mathrm{U})$, where $g(.)$ denotes the perceived quality of the rendered sequence.

4.4. A graphical interpretation of the results

A graphical interpretation of the experimental results can be made as follows. In Fig. 15, let the horizontal axis denote the amount of average spatial segmentation errors (i.e. accuracy) and let the vertical axis denote the temporal variation of spatial errors. We denote the iso-perceptual

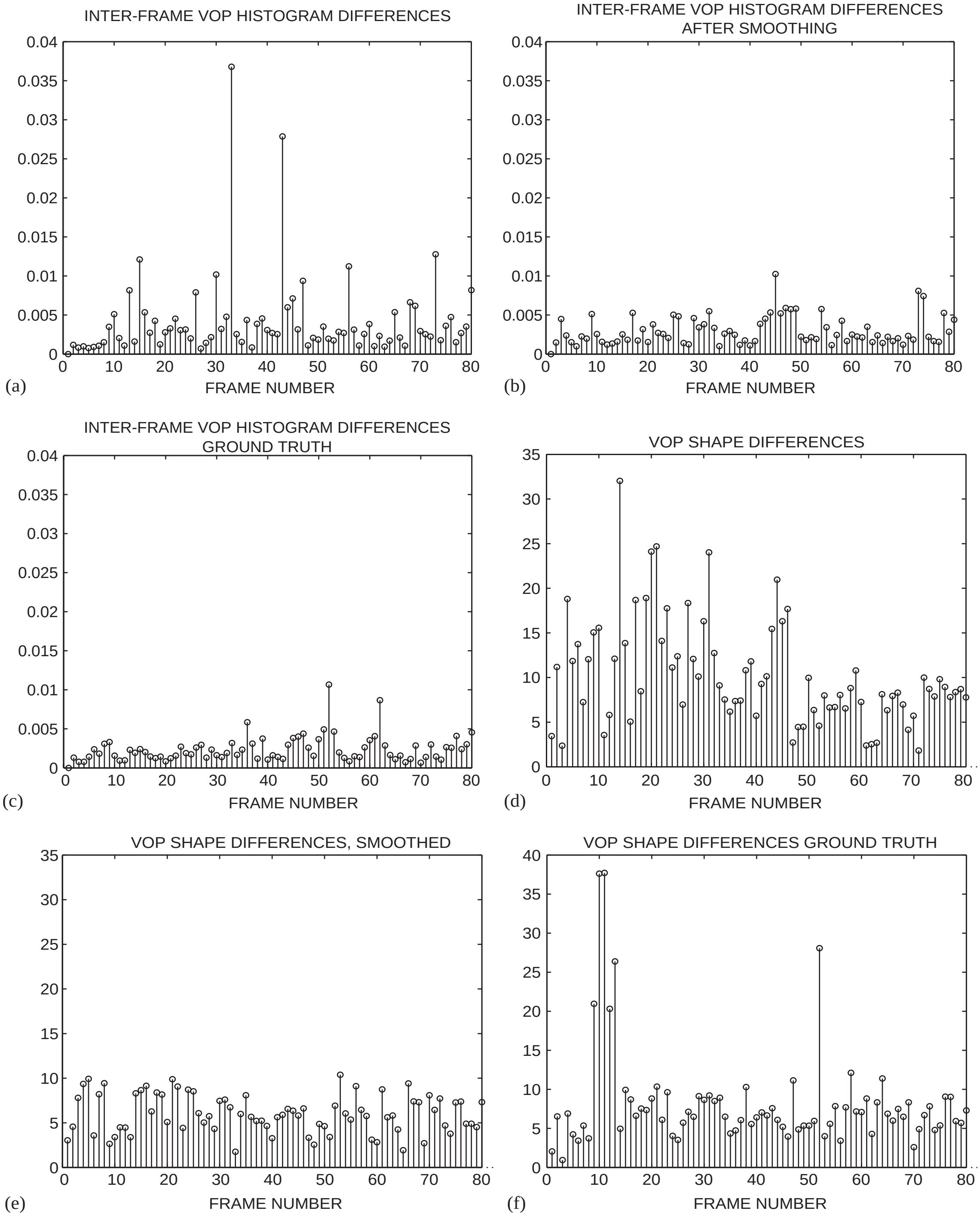


Fig. 14. Man object: (a), (b): The histogram difference measure before (a) and after (b) spatio-temporal stabilization versus frame number. (d), (e): The shape difference measure before (d) and after (e) spatio-temporal stabilization.

Table 2
Objective evaluation scores for the man object before and after spatio-temporal smoothing

	Histogram measure		Shape measure	
	Mean ($\times 10^{-3}$)	Variance ($\times 10^{-6}$)	Mean	Variance
Before smoothing	4.17	28.3	10.61	43.76
After smoothing	2.97	3.38	6.3	8.66
Ratio (before/after)	1.4	8.37	1.68	5.05
Ground-truth	2.37	2.74	8.15	41.43

Table 3
Subjective evaluation scores for the Flikken sequence

	Tests 1–2	Tests 3–4	Tests 5–7	Tests 8–9
Viewed AB pair	-HU, UH	SH, -HS	UU, HH, SS	-SU, US
Average score	1.05	0.59	0.08	0.52

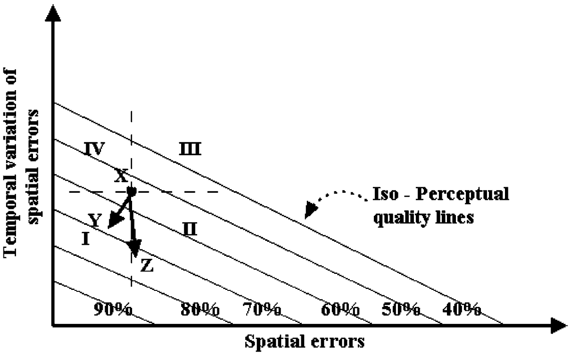

Temporal variation of spatial errors
IV
III
X
Iso - Perceptual quality lines
Y
I
II
Z
90%
80%
70%
60%
50%
40%
Spatial errors

Fig. 15. A graphical interpretation of the results.

quality points by slanted lines, assuming that less spatial segmentation accuracy can be compensated by less temporal variation of spatial errors. According to this illustration, a perfect (ground-truth) segmentation will be at the origin, with no spatial errors, no variation of spatial errors and perceptually perfect. We denote the given temporally unstable segmentation results of an algorithm by the point X. In order to increase the quality of the given segmentation X, the ideal direction to go would be the direction denoted by Y, in which the spatial errors and temporal variation of spatial errors decrease (temporal stability increases), and perceptual quality increases. However, achieving this is generally more difficult, if not impossible, as compared to going in the direction Z, which keeps the amount of spatial errors almost the same, but increases the temporal stability and hence the perceptual quality. This is actually the direction taken by the algorithm presented in this paper. In fact, going in any direction (in quadrant II) is better than staying at the point X, as long as the perceptual quality increases. We have shown by the experimental results that it is indeed possible to increase the perceptual quality of object segmentation in 3D-TV applications by increasing temporal stability, while keeping the average spatial errors almost the same.

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 curve evolution technique for temporal stabilization of video object segmentation 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 perceptual object segmentation quality without increasing the segmentation accuracy. An object segmentation algorithm which optimizes the temporal stability measures directly is under development.

Acknowledgements

We would like to thank Roos Rajae and Ingrid Heynderickx for the useful discussions during perceptual evaluation tests.

References

[1] E.M. Arkin, L.P. Chew, D.P. Huttenlocker, K. Kedem, J.S.B. Mitchell, An efficient computable metric for comparing polygonal shapes, IEEE Trans. Pattern Anal. Mach. Intell. 13 (1991) 209–215.

[2] G. Aubert, M. Barlaud, O. Faugeras, S.J. Besson, Image segmentation using active contours: calculus of variations or shape gradients? SIAM J. Appl. Math. 63 (6) (2002) 2128–2154.

[3] A.M. Baumberg, Learning deformable models for tracking human motion, Ph.D. Thesis, The University of Leeds, School of Computer Studies, October 1995.

[4] S.J. Besson, M. Barlaud, Dreams: deformable regions driven by an eulerian accurate minimization method for image and video segmentation, Int. J. Comput. Vision 53 (1) (2003) 45–70.

[5] A. Blake, M. Isard, Active Contours, Springer, Berlin, 1998.

[6] C. Bregler, J. Malik, Tracking people with twists and exponential maps, Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, June 1998, pp. 239–245.

[7] T. Cham, J.M. Regh, A multiple hypothesis approach to figure tracking, Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 1999, pp. 239–245.

[8] P.L. Correia, F.P. Pereira, Objective evaluation of video segmentation quality, IEEE Trans. Image Process. 12 (2) (February 2003) 186–200.

[9] G. Engeldrum, Psychometric Scaling: A Toolkit for Imaging Systems Development, Imcotek Press, 2000.

[10] C.E. Erdem, A.M. Tekalp, B. Sankur, Metrics for performance evaluation of video object segmentation and tracking without ground-truth, Proceedings of IEEE International Conference on Image Processing (ICIP), vol. 2, 7–10 October, Greece, 2001, pp. 69–72.

[11] C.E. Erdem, A.M. Tekalp, B. Sankur, Video object tracking with feedback of performance measures, IEEE Trans. Circuits Systems Video Technol. 13 (4) 2003.

[12] C.E. Erdem, B. Sankur, A.M. Tekalp, Performance measures for video object segmentation and tracking, IEEE Trans. Image Process. 13 (937–951) (2004) 195–216.

[13] F. Ernst, 2d-to-3d video conversion based on time-consistent segmentation, Proceedings of the ICOB'03 Workshop, 2003.

[14] F. Ernst, P. Wilinski, K. van Overveld, Dense structure-from-motion: an approach based on segment matching, Proceedings of European Conference on Computer Vision, 2002.

[15] R. Feghali, A. Mitiche, Tracking with simultaneous camera motion subtraction by level set spatio-temporal surface evolution, Proceedings of IEEE International Conference on Image Processing (ICIP), 2003.

[16] B. Goldlucke, M.A. Magnor, Joint 3d-reconstruction and background separation in multiple views using graph cuts, Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, vol. 1, 2003, pp. 683–688.

[17] M. Isard, A. Blake, Condensation-conditional density propagation for visual tracking, Int. J. Comput. Vision 29 (1) (1998) 5–28.

[18] S.X. Ju, M.J. Black, Y. Yacoob, Cardboard people: a parametrized model of articulated image motion, Proceedings of the International Conference on Face and Gesture Recognition, 1996, pp. 561–567.

[19] A. Mansouri, A. Mitiche, M. Aron, Pde-based region tracking without motion computation by joint space-time segmentation, Proceedings of IEEE International Conference on Image Processing (ICIP), 2003.

[20] M. Op de Beeck, A. Redert, Three dimensional video for the home, Proceedings of the International Conference on Augmented Virtual Environments and Three-Dimensional Imaging, 2001, pp. 188–191.

[21] B. Scassellati, S. Alexopoulos, M. Flickner, Retrieving images by 2d shape: a comparison of computation methods with human perceptual judgements, Proceedings of the SPIE Conference on storage and Retrieval for Image and Video Database, vol. 2185, 1994, pp. 2–14.

[22] D. Scharstein, R. Szeliski, High-accuracy stereo depth maps using structured light, Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, vol. 1, 2003, pp. 195–202.

[23] P. Smith, T. Drummond, R. Cipolla, Edge tracking for motion segmentation and depth ordering, Proceedings of the 10th British Machine Vision Conference, vol. 2, 1999, pp. 369–378.

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

[25] S. Wachter, H.H. Nagel, Tracking persons in monocular image sequences, Comput. Vision Image Understanding 74 (3) (June 1999) 174–192.

[26] A. Yezzi, A. Tsai, A. Willsky, A fully global approach to image segmentation via coupled curve evolution equations, J. Visual Commun. Image Represent. 13 (2002) 195–216.

[27] S. C. Zhu, A. Yuille, Region competition: unifying snakes, region growing, and bayes/mdl for multiband image segmentation, IEEE Trans. Pattern Anal. Mach. Intell 18 (9) (1996).