Multiple-Feature Spatiotemporal Segmentation of Moving Sequences using a Rule-based Approach

Roberta Piroddi and Theodore Vlachos
Centre for Vision, Speech and Signal Processing,
University of Surrey
Guildford, GU2 7XH, UK
r.piroddi,t.vlachos@surrey.ac.uk

Abstract

In this paper a novel two-stage architecture for object-based segmentation of moving sequences is proposed using multiple features such as motion, intensity and texture. The first stage locates perceptually meaningful objects using a hierarchy of single-feature segmentation processes. The second stage refines the boundaries of located objects using a combination of features according to a set of appropriate rules. Experimental results show that the proposed approach yields intuitively correct as well as accurate segmentations of moving sequences, which compare favourably with established state-of-the art techniques in the literature.

1 Introduction

Emerging multimedia applications and services require efficient and flexible coding (MPEG-4) and description (MPEG-7) of visual information. Object-based representations of visual information are particularly suited to this purpose, since they endeavour to describe a dynamic scene in terms of its constituents instead of low-level visual primitives, such as intensity, colour, texture, motion, depth and shape [1].

Moving object segmentation techniques can be divided into three categories, depending on the kind of feature space employed for the segmentation: motion-based segmentation techniques, spatio-temporal segmentation techniques and multiple-feature segmentation techniques [2].

Motion-based segmentation techniques can be further divided into two categories: top-down and bottom-up approaches [3]. Top-down approaches rely on the iterative estimation of global motion and the rejection of outliers for successive elaboration [4][5]. Bottom-up approaches [6] rely on region-merging paradigms which employ suitable similarity functions. Motion-only segmentation techniques are well suited to video coding. However they fail to detect meaningful objects accurately, due to excessive reliance of local neighbourhood processing, which is commonly employed by motion estimation algorithms to improve their robustness.

BMVC 2002 doi:10.5244/C.16.33

In order to obtain accurate identification of object boundaries, spatio-temporal segmentation techniques operate in the spatial (intra-frame) and temporal (inter-frame) domain, using intensity/colour and motion information respectively. These methods can be further classified into two different categories according to the strategy employed to fuse spatial and temporal information: hierarchical spatio-temporal segmentation methods and parallel spatio-temporal segmentation methods. Hierarchical methods rely on the definition of joint spatio-temporal similarity (or homogeneity) criteria [3][7]. On the basis of such criteria, they proceed in a region-merging fashion. Joint spatio-temporal similarity criteria usually require weights, that are sequence dependent and often based on heuristics. They often result in over-segmentation and are computationally expensive. On the other hand, parallel methods [8][9] first perform single-feature spatial and temporal segmentations separately and then successively integrate the resulting partitions employing an appropriate set of rules. These methods manage to overcome the over-segmentation problems of hierarchical techniques, achieve good accuracy in the location of object boundaries and are simpler to implement.

There is evidence that simultaneous consideration of a number of features, such as intensity, motion, texture, colour, depth, and so on, is useful towards reinforcing the spatio-temporal coherence of objects, which the above methods fail to achieve. Most recent attempts to perform multiple-feature segmentation use clustering, neural networks and supervised user intervention [10][11][12]. These approaches have similar problems as the hierarchical spatio-temporal techniques, namely over-segmentation, reliance on weighting strategies and computational complexity. Parallel use of features as used by parallel spatio-temporal techniques has shown to overcome these problems [8].

In this paper we present a novel architecture for object-based segmentation of moving sequences. This architecture comprises two stages. In the first stage objects of interest are located using single features and exploiting global information to avoid over-segmentation. The features used are motion, intensity and texture. In the second stage a region-merging rule-based approach is used, employing multiple features and exploiting local information across object boundaries. The proposed architecture has a number of important advantages. It combines the best features from parallel spatio-temporal, top-down and bottom-up approaches such as the efficiency of rule-based integration, the reliability of global information and the accuracy of local information respectively.

This paper is structured as follows. In Section 2 an outline of the proposed architecture is given: in 2.1 the first stage of the architecture is described, in 2.2 the second stage of the architecture is described with more details about the rule-based processor given in 2.2.3. In Section 3 experimental results are presented and evaluated using state-of-the-art moving object segmentation methods and conclusions are drawn.

2 System Architecture

The proposed architecture is an evolution of the two-stage approach presented in [13]. We present an improved version of the algorithm, where significant benefits in terms of efficiency are achieved due to the use of mathematical morphology. An outline of the algorithm is shown in Figure 1 and its main features are re-iterated below.

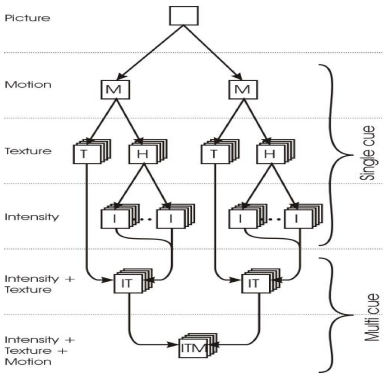


Figure 1: Flowchart of proposed architecture.

2.1 Construction of the Feature Hierarchy

This first stage comprises the following three layers.

2.1.1 Motion segmentation layer (M-layer)

Image F (a frame of a moving sequence at time t) is segmented into several non-overlapping and collectively exhaustive motion segments M_i so that $F = \bigcup_{i=1}^{N} M_i$ and $M_i \bigcap M_j = \emptyset$, when $i \neq j$, for $i = 1, \ldots, N$, N being the total number of motion segments. Motion segmentation is obtained using a region growing method based on a generalised version of the Recursive Shortest Spanning Tree (RSST) algorithm [14][15]. Regions are formed by iteratively merging the two most similar vertices in the graph according to a given cost function. A feature vector is initially assigned to a vertex consisting of the horizontal and vertical components $[u(p)v(p)]$ of a previously computed dense optic flow field [4]. After k iterations the mean optic flow components computed $[\bar{u}(p)\bar{v}(p)]$ over a region p are assigned to vertex $\vec{V^k}(p)$, so that $\vec{V^k}(p) = [\bar{u}(p)\bar{v}(p)]^T$. The cost function between two regions p_i and p_j used in this case is:

$$C_m(p_i, p_j) = |\vec{V^k}(p_i) - \vec{V^k}(p_j)| \tag{1}$$

and $|\cdot|$ is the L_1 norm.

2.1.2 Texture segmentation layer (T-layer)

Each motion segment M_i is further segmented using textural activity criteria. Segments of high textural activity corresponding to motion segment M_i will be indicated as T_i^l, where $T_i^l \subseteq M_i$ for $l = 1, \ldots, L$, while segments of low textural activity will be indicated as H_i^m where $H_i^m \subseteq M_i$ for $m = 1, \ldots, M$ and $\{\bigcup_{l=1}^{L} T_i^l\} \bigcup \{\bigcup_{m=1}^{M} H_i^m\} = M_i$. We adopt texture activity criteria based on mathematical morphology. This leads to a substantial improvement over previously reported work [13] and for this reason it is presented in more detail below.

The accurate location of textured areas of an image plays a fundamental role in moving-object segmentation for two main reasons. In the temporal part of the segmentation, the reliability of motion information depends strongly on textural content of the region of support [12]. In the spatial part of the segmentation, it avoids over-segmentation and contributes to produce perceptually consistent results.

Several approaches have been developed for texture analysis for still images [17]. Examples are Normalized Cuts, relaxation techniques, Markov Random Fields [19][18] and so on. Unfortunately these can be time consuming and/or tractable only if the number of segmentation labels is limited and known in advance. Due to the volume of data to be processed, texture analysis for video processing must be fast and efficient. We adopt an approach based on mathematical morphology [20], which is very fast and efficient to implement. Moreover, it does not require the use of windows for the evaluation of features, allowing for an extremely accurate location of edges of the textured areas.

The textural analysis starts with the extraction of edges, using the Canny [20] edge detector. In mathematical morphology, the combination of dilation followed by erosion with the use of the same structural element is called closing operation. The dilation extends the support of isolated edges until they merge with adjacent open contours. This produces a region of support locating a textured area. The erosion operation that follows restores the real boundaries of the textured areas while avoiding strong isolated edges to be misinterpreted for textured.

The results of the above procedure are presented in Figure 2. Figure 2.a) shows texture segmentation obtained using an edge-density measure followed by hard thresholding as proposed in [13]. Figure 2.b) shows segmentation obtained using mathematical morphology with edges highlighted in white. We note the improved accuracy in the location of the boundaries of perceptually meaningful textured areas.

2.1.3 Intensity segmentation layer (I-layer)

Each low-texture segment H_i^m is subject to grey-level segmentation, using the RSST algorithm. In this case the feature vector after k iterations is the mean grey-level intensity $\bar{i}(p)$ computed over a region p. The cost function used is:

$$C_i(p_i, p_j) = |\bar{i}(p_i) - \bar{i}(p_j)| \tag{2}$$

The set of segments produced by this segmentation is indicated as $I_i^{m,j}$ so it holds that $\bigcup_{j=1}^{J} I_i^{m,j} = H_i^m$ and $\{\bigcup_{l=1}^{L} T_i^l\} \bigcup \{\bigcup_{m=1}^{M} \bigcup_{j=1}^{J} I_i^j\} = M_i$.

2.2 Refinement of Boundaries

As already stated, the first stage produces rather inaccurate boundaries which need improvement. This is accomplished at this stage, where processing is restricted to the neighbourhood of the boundaries, thereby reducing the amount of information to be processed. The second stage comprises two layers.

2.2.1 Intra-M-Feature boundary refinement (IT-layer)

Within the boundaries of the same motion segment M_i, segments $I_i^{m,j} \subseteq M_i$ and $T_i^l \subseteq M_i$ are merged using the RSST algorithm. For this purpose each segment is specified by

its average intensity. The cost function used in this case is given by equation 2. A segment obtained as above will be denoted as S_i^r, for $r = 1, \ldots, R$.

2.2.2 Inter-M-Feature boundary refinement (ITM-layer)

We define B_j the common boundary between two motion segments M_i and M_j.

We define (S_i^r, S_j^q) a pair of segments that touch a common boundary and lie on different motion segments so that $S_i^r \subseteq M_i$ and $S_j^q \subseteq M_j$. We therefore define M_j as the *opposite* motion segment for S_i^r, since S_i^r touches the motion segment M_j on a common boundary $B_{i,j}$, without belonging to it. Vice-versa M_i is defined as the *opposite* motion segment for S_j^q. Each pair of segments lying on a boundary between motion segments is evaluated and merged if appropriate according to a set of given rules, detailed below.

2.2.3 Rule-based processing

The refinement of common boundaries between two motion segments M_i and M_j is carried out using an appropriate set of rules. Extending the work in [13], we present a simplified set of rules.

Each segment S_i^r lying on a boundary $B_{i,j}$ is specified by three features: average optic flow in the horizontal U_i^r and vertical direction V_i^r, and average intensity I_i^r . We denote the average horizontal and vertical optic flow of each motion segment M_i as U_i and V_i respectively. If the area of a segment S_i^r is A_i^r, we define: $min(S_i^r, S_j^q) = min(A_i^r, A_j^q)$. We also define the operation of merging a segment S_i^r with its *opposite* motion segment M_j as $merge(S_i^r)$.

The motivation for this rule-based approach is that optic flow is more reliable if the regions have a high textural activity; otherwise joint consideration of spatial and temporal features is needed. The application of these rules requires thresholds as follows: P_i against which the intensity difference between two segments is compared, and P_s against which region size is compared, in order to remove small regions. Threshold selection is discussed further in Section 3. The above rules can be described by the following pseudo-code, where l_i is the number of textured segments belonging to M_i and l_j is the number of textured segments belonging to M_j:

if $((S_i^r \not\subseteq T_i^{l_i}) \vee (S_j^q \not\subseteq T_j^{l_j}))$

- if $(|I_i^r - I_j^q| \leq P_i)$
  - if $((A_i^r \leq T_s) \vee (A_j^q \leq P_s))$
    - $merge(min(S_i^r, S_j^q))$
  - else
    - if$((|U_i^r - U_j| + |V_i^r - V_j|) < (|U_i^r - U_i| + |V_i^r - V_i|))$
      - $merge(S_i^r)$
    - if$((|U_j^q - U_i| + |V_j^q - V_i|) < (|U_j^q - U_j| + |V_j^q - V_j|))$
      - $merge(S_j^q)$

If segment S_i^r has neighbours belonging to more than one motion segment M_i the above rules are modified accordingly.

3 Experimental results and discussion

Experimental results using the test sequences *Renata*, *Garden* and *Mobile and Calendar* are presented. As mentioned in Section 2.2, two thresholds need to be specified. P_s is expressed as a percentage of the largest segment in the image and has be set to 5% in all sequences here presented. P_i is a measure of the intensity difference, or contrast, between different segments. It is chosen in such a way that $P_i < \bar{P}_i$, where $\bar{P}_i$ is the average intensity difference according to (2) among all segments at the *ITM-layer*. The value of P_i has been set equal to 32 for *Renata*, 24 for *Mobile and Calendar* and 10 for *Garden*. With regards to the morphological texture segmentation, all the experimental results presented here are obtained by using the same structural element and the same settings for the Canny filter (i.e. the default settings in Matlab 6.1). The structuring element is a flat disk-shaped element with radius 3 and 4-connected neighbourhoods are considered. In [13] a third threshold P_t was used to discriminate between *textured* and *not textured* regions of the image, but this was difficult to determine and ultimately user-dependent. In this work, the need for P_t has been eliminated.

In Figure 3 the segmentation of a frame of *Renata* is shown. In 3.a) the output of the *M-layer* is shown, with region boundaries superimposed on the original image. In 3.b) the output of the *T-layer* is shown. In 3.c) the segments obtained from the *IT-layer* are shown. In 3.d) the final segmentation obtained from the *ITM-layer* is shown. From the comparison between 3.a) and 3.d) it is possible to notice the improvement obtained in the definition of the boundaries with respect to the motion segmentation, especially in regions with low texture (like the arm and the wall) where the motion information is not very reliable. The algorithm avoids over-segmentation of the image; this is particularly evident in the segmentation of highly textured objects (like the calendar and the tapestry). In Figure 4 the segmentation of the first four frames of the test sequence of *Mobile and Calendar* is shown. This sequence consists of multiple moving objects. Object boundaries are located consistently from frame to frame, without explicit need for segment tracking.

Comparative assessment of results is carried out considering established sequence segmentation methods. Different segmentations are shown using frame 10 of the MPEG test sequences *Renata* and *Garden*.

The first method considered is a top-down approach to motion-only segmentation [4]. A dense optic flow is evaluated using a robust multi-resolution motion estimation method. Then the global motion components are extracted in order to obtain a change detection mask. The results are shown in Figure 5.a) for *Renata* and Figure 5.b) for *Garden*. This method seems effective towards locating semantically meaningful object and the results are consistent during a long sequence. However, the inaccuracy in the determination of real boundaries, makes it less suitable for content-driven analysis.

The second method adopts a bottom-up approach to motion-based segmentation using a layered representation of information as discussed in [6]. This method is capable of extracting motion planes. This is particularly useful for video coding applications. However it fails to locate perceptually meaningful objects. Moreover, due to over-segmentation of the regions corresponding to moving objects, the located boundaries are inconsistent over time. Results obtained using this method are shown in Figure 5.c) for *Renata* and Figure 5.d) for *Garden*.

The third method is a hierarchical spatio-temporal technique where temporal and spatial information is integrated into a single similarity function [7]. Motion and intensity

similarity contribute to this function using a weighting scheme often employed in fuzzy clustering. The iterative region merging is done evaluating the joint similarity of neighbouring regions using the watershed segmentation technique. The results of the application of this method are shown in Figure 5.e) for *Renata* and Figure 5.f) for *Garden*. This method suffers also from a degree of spatial over-segmentation. Moreover, due to the sensitivity of the watershed transform to noise, the tracked boundaries are often not consistent throughout a long sequence.

The fourth method is a parallel spatio-temporal segmentation technique which uses single-feature temporal and spatial segmentation in order to obtain spatial and temporal masks and then combines the results obtained using a set of appropriate rules [8]. The results of the application of this method are shown in Figure 5.g) for *Renata* and Figure 5.h) for *Garden*.

Finally, the segmentation results obtained using the proposed method are presented in Figure 5.i) for *Renata* and Figure 5.l) for *Garden*. There is a notable improvement in the accuracy of the boundaries obtained, even in regions of low contrast, in which the two spatio-temporal methods fail. This is due to the progressive refinement achieved by the rule-based approach that combines single-feature segmentations. The perceptually meaningful objects are located accurately and this makes this method particularly suitable for multimedia content-driven applications.

Overall our results demonstrate that the proposed algorithm yields intuitively correct segmentations corresponding to actual scene objects whose boundaries have been extracted with a substantial degree of accuracy. Furthermore, our comparative assessment suggests that our segmentation method compares favourably with established state-of-the-art moving sequence segmentation algorithms.

4 Conclusions

In this paper, a novel architecture for object-based segmentation of moving sequences using multiple features has been proposed. The architecture comprises two stages of processing. The first stage produces a hierarchy of segments that are extracted using single features, namely motion, texture and intensity. The second stage fuses multiple features to effectively reinforce the spatio-temporal coherence of the segments obtained in the first stage. We adopted an effective morphological texture segmentation technique which allows for the accurate definition of region boundaries and requires low-complexity.

The proposed technique represents a significant improvement in the field of object-oriented segmentation as it incorporates in a unique framework favourable features of both hierarchical and parallel spatio-temporal segmentation techniques. A further useful feature is the capability to achieve a hierarchical decomposition of a scene into perceptually significant objects; not only moving objects are accurately located, but also texture segmentation is fast and reliable. This hierarchical representation of the moving scene can be exploited towards object-based analysis and video coding.

The technique is attractive from an implementation point of view, in the sense that it avoids computationally expensive homogeneity criteria and weighting functions that competing methods require, while most processing stages can be carried out in parallel.

References

[1] P. Salembier, F. Marques, *"Region-based representations of image and video : Segmentation tools for multimedia services,"* IEEE Trans. Circ. Sys. Vid. Tech., Vol. 9(8), 1999, pp. 1147-1167.

[2] D. Zhang, G. Lu, *"Segmentation of Moving Objects in Image Sequence: A Review,"* Circuits, Systems and Signal Processing, Vol. 20(2), 2001, pp. 143-183.

[3] F. Moscheni, S. Batthacharjee, M. Kunt, *"Spatiotemporal Segmentation Based on Region Merging,"* IEEE Trans. PAMI, Vol. 20(9), 1998, pp. 897-914.

[4] M.J. Black, P. Anandan, *"The robust estimation of multiple motions: parametric and piecewise-smooth flow fields,"* Computer Vision and Image Understanding, Vol. 63(1), 1996, pp. 75-104.

[5] K. Zhang, M. Bober, J. Kittler, *"Motion based image segmentation for video coding,"* Proc. ICIP, Vol.3, 1995, pp.476-479.

[6] J.Y.A. Wang, E.H. Adelson, *"Representing moving images with layers," IEEE Trans. Image Processing,"* Vol. 3(5), 1994, pp. 625-638.

[7] J.G. Choi, S.-W. Lee, S.-D. Kim, *Video segmentation based on spatial and temporal information,"* IEEE ICASSP, Vol.4, 1997, pp.2661-2664.

[8] A.A. Alatan, L. Onural, M. Wollborn, R. Mech, E. Tuncel, T. Sikora, *"Image sequence analysis for emerging interactive multimedia services-the European COST 211 framework,"* IEEE Trans. Circ. Sys. Vid. Tech., Vol. 8(7), 1998, pp. 802-813.

[9] D. Wang, *"Unsupervised video segmentation based on watersheds and temporal tracking,"* IEEE Trans. Circ. Sys. Vid. Tech., Vol. 8(5), 1998, pp. 539-546.

[10] R. Castagno, T. Ebrahimi, M. Kunt, *"Video segmentation based on multiple features for interactive multimedia applications,"* IEEE Trans. Circ. Sys. Vid. Tech., Vol. 8(5), 1998, pp. 562-571.

[11] C. Gu, M.C. Lee, *"Semiautomatic Segmentation and tracking of semantic video objects,"* IEEE Trans. Circ. Sys. Vid. Tech ,Vol. 8(3), 1998, pp. 572-584.

[12] J. Kim, T. Chen, *"Low-complexity fusion of intensity, motion, texture, and edge for image sequence segmentation: a neural network approach,"*, Proc. Neural Networks for Signal Processing X, Vol.2, 2000, pp. 497-506.

[13] R. Piroddi, T. Vlachos, *"Object-based Segmentation of Moving Sequences using Multiple Features," to appear* in Proc. Int. Conf. Digital Signal Processing, 2002.

[14] O.J. Morris, M.deJ. Lee, A.G. Constantinides, *"Graph theory for image analysis: an approach based on the shortest spanning tree,"* IEE Proceedings, Vol. F-133(2), 1986, pp.146-152.

[15] T. Vlachos, A.G. Constantinides, *"A graph-theoretic approach to colour image segmentation and contour classification,"* Proc. Int. Conf. Image Processing and its Applications , 1992, pp.298-302.

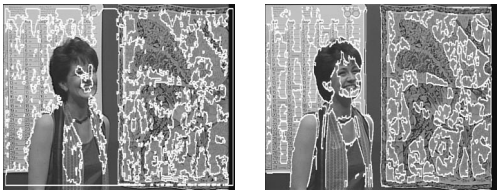
[16] B.S. Manjunath, J-R. Ohm, V.V. Vasudevan, A. Yamada, *"Color and texture descriptors,"* IEEE Trans. Circ. Sys. Vid. Tech., Vol. 11(6),2001, pp. 703-715.

[17] M. Mirmehdi, M. Petrou, *"Segmentation of color textures,"* IEEE Trans. PAMI, Vol. 22(2), 2000, pp. 142-159.

[18] J. Malik, S. Belongie, T. Leung, J. Shi, *"Contour and texture analysis for image segmentation,"* Int. Jour. Computer Vision, Vol. 43(1) , 2001, pp. 7-27.

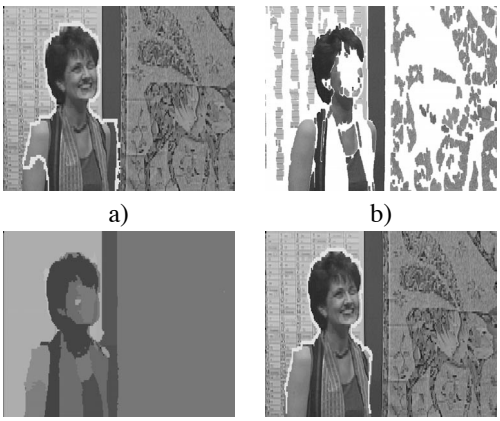
[19] E. Liapis, E. Sifakis, G. Tziritas, *"Color and/or Texture Segmentation using deterministic Relaxation and fast marching algorithms,"* Proc.ICPR, Vol. 3, 2000, pp. 621-624.

[20] M. Sonka, V. Hlavac, R. Boyle *"Image Processing, Analysis, and Machine Vision,"*, PWS Publishing, Second edition, 1999.



a) b)

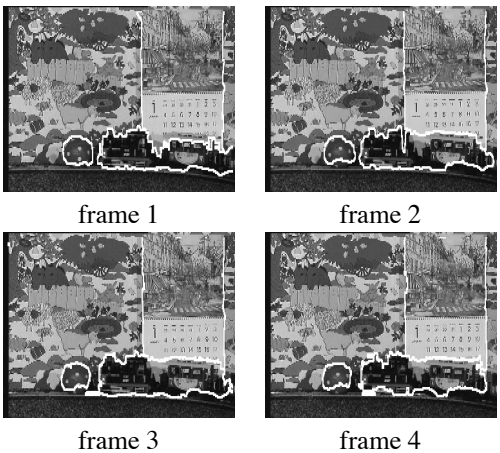
Figure 2: Texture Segmentation of test sequence *Renata*: a) hard-thresholding of edge density, b) segmentation using mathematical morphology.



a) b)

c) d)

Figure 3: Segmentation of test sequence *Renata*: a) *M-layer* segmentation, b) *T-layer* segmentation, c) *IT-layer* segmentation, d) *ITM-layer* segmentation.



frame 1 frame 2

frame 3 frame 4

Figure 4: Segmentation of test sequence *Mobile and Calendar* for frames 1, 2, 3 and 4.

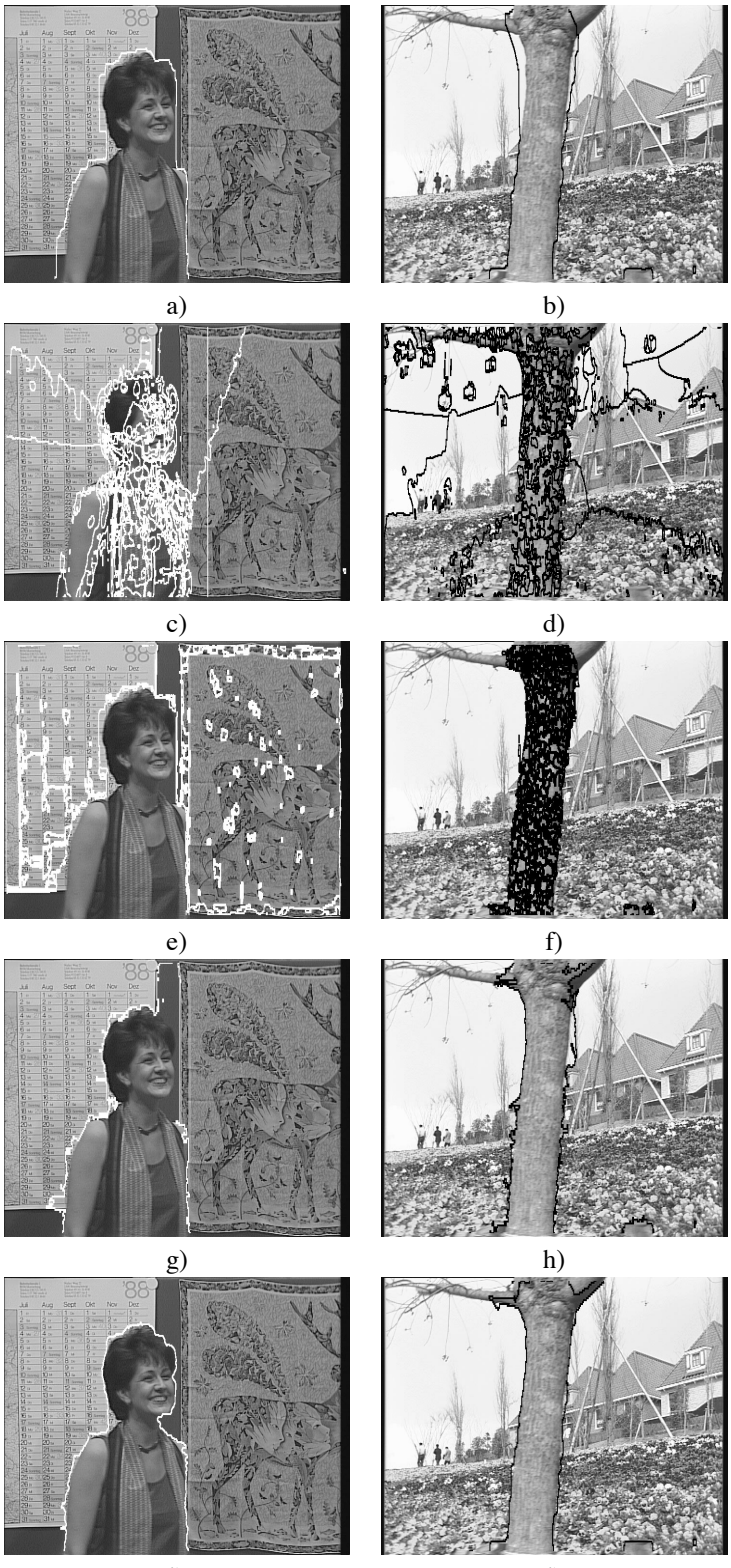


Figure 5: Qualitative comparison between moving object segmentation techniques: motion-based top-down approach [4] a) and b), motion-based layered representation [6] c) and d), joint similarity spatio-temporal segmentation [7] e) and f), rule-based spatio-temporal segmentation [8] g) and h), method presented in this paper i) and j).