Video Objects Segmentation by Robust Background Modeling

Andrea Colombari, Andrea Fusiello, and Vittorio Murino
Dipartimento di Informatica, Università degli Studi di Verona
Strada Le Grazie, 15 - 37134 Verona, Italy
colombari@sci.univr.it, {andrea.fusiello,vittorio.murino}@univr.it

Abstract

This paper deals with the problem of segmenting a video shot into a background (still) mosaic and one or more foreground moving objects. The method is based on ego-motion compensation and background estimation. In order to be able to cope with sequences where occluding objects persist in the same position for a considerable portion of time, the papers concentrates on robust background estimation method. First the sequence is subdivided in patches that are clustered along the time-line in order to narrow down the number of background candidates. Then the background is grown incrementally by selecting at each step the best continuation of the current background, according to the principles of visual grouping. The method rests on sound principles in all its stages, and only few, intelligible parameters are needed. Experiments with real sequences illustrate the approach.

1. Introduction

The usefulness of digital video – which is nowadays widespread on the World Wide Web and in multimedia databases – is limited by the lack of a true content-based description, that would allow interactive manipulation and adaptation. Content-based representation of videos was introduced several years ago in the MPEG-4 [15] standard, yet reliable and automatic tools for automatic extraction of Video Objects are not available. Although some attempts have been made [10, 5, 31, 18, 4], the challenge is to cope with real, complex situations, where Video Objects interacts with themselves and with the environment.

This paper makes a step forward in this direction, in the case where the scene is composed by a static background plus some foreground (possibly moving) objects.

After having compensated for camera motion, foreground objects can be extracted effectively by subtracting the static background from each frame [23], provided that the background can be estimated. The problem – also called *background initialization* in the surveillance literature – is defined as follows: given a video sequence taken with a stationary camera, in which a static background is occluded by any number of foreground moving objects, output a single image of the static background (even if such an image have never been captured).

In the most fortuitous cases, foreground has the property to insist on each pixel location for less than 50% of the entire sequence length. In this case background is obtained as the median of each pixel color distribution. Other techniques [26, 8, 30] have been proposed which, like the median, operate at pixel-level, making decisions independently for each pixel. The Adaptive Smoothness Method [17], for example, finds intervals of stable intensity in the sequence. Then, using some heuristics, the longest stable value for each pixel is selected and used as the value that most likely represents the background. Unfortunately, pixel-level data can be useful in narrowing the number of possible candidate values for the background, but, if foreground is stationary for a long period of time, these techniques fail.

Spatial support must be taken into consideration as an additional heuristics in order to overcome this problem [19, 12, 16]. The Local Image Flow algorithm [6], for instance, considers also information generated by the neighboring locations, namely the local optical flow. Background values hypotheses are generated by locating intervals of relatively constant intensity, which are weighted with local motion information. This technique, however, cannot cope with foreground objects that move only in few frames, or equivalently, with the problem of estimating the background from two images only.

Our approach is able to cope with sequences where foreground objects persist in the same position for a considerable portion of time. First the sequence is subdivided in patches that are clustered along the time-line in order to narrow down the number of background candidates. Then the background is grown incrementally by selecting at each step the best continuation of the current background. Spatial continuity is enforced through the principles of visual grouping [28].

Related works can be found in the areas of *video inpainting* [29, 20, 13] where the problem is to repair holes in a video sequence with plausible values. Background initialization could be cast as video inpainting if the foreground masks were known beforehand, which do not makes sense in our case. Moreover we seek to estimate a physically valid view of the background, by choosing pixel values only along the same time-line, whereas this is not usually a constraint in video inpainting.

Regarding background initialization, the closest works to ours are [2] and [22], that deals with background initialization and mosaic completion respectively. They are based on the same scheme: (i) identify an initial region which is sure to be background and then (ii) fill-in the remaining unknown background incrementally by choosing values from the same time-line. At each step, the patch that maximizes a likelihood measure with respect to the surrounding zone, already identified as background, is selected. This entails that the background should be self-similar (like a building's facade) and that the starting region should be large enough to provide sufficient information. On the contrary, this need not to be assumed in our algorithm.

2. Overview

The input is a video shot[1] with a stationary background. If the background is planar (like a facade) the camera can move freely, otherwise it is constrained to rotate only (like in a panning operation). These constraints derive from the fact that background recovery is based on mosaicing.

The output is a representation of the sequence suitable to be encoded in MPEG-4 (Main Profile) [15]. The central concept in MPEG-4 is that of the Video Object (VO). Being content-based oriented, MPEG-4 considers a scene to be composed of several VOs, which are separately encoded [21]. Each VO is characterized by intrinsic properties such as shape, texture, and motion. Shape is represented by a binary mask or by an 8-bit transparency mask (this feature is available in the MPEG-4 Main Profile).

In our case, the shot is represented as being composed by a *sprite panorama* (i.e., a still image describing the content of the background over all the frames in the shot) and one arbitrary-shape VO for each foreground object, with a binary mask as shape descriptor. For each frame, the global motion parameters are given by the coordinates of the four corners of the image transformed in the mosaic reference frame.

The method we are proposing for extracting Video Objects is based on segmenting moving objects from the static background and tracking them in the video sequence. As the processing is non-causal, all the frames composing the video shot are needed simultaneously.

The processing pipeline can be decomposed into several stages: Ego-motion compensation (Sec. 3), Background modeling (Sec. 4), Foreground segmentation (Sec. 5), and Blob tracking (described elsewhere [4]), where blobs are tracked through the sequence – using frame-to-frame matching and a graph representation – and associated to Video Objects. The overall framework was set forth in [18], but the background estimation and the blob tracking were extremely simple and relied on fairly restrictive assumptions. In [4] we improved radically the blob tracking algorithm, allowing for occlusions between objects, occlusions between an object and background, objects entering and leaving the scene at any point. In [3] we proposed the robust technique for background recovery that is also described here. The experiments reported in Sec. 6 have been performed with the whole pipeline, including tracking.

3. Ego-motion compensation

Camera's motion compensation is carried out with respect to a *stabilization* plane, which can be either a scene plane or the infinity plane. In the latter case camera's motion is constrained to rotate and all the static components of the scene are stabilized. In the former case, instead, even static objects that do not lay on the stabilization plane exhibit a residual motion, called *parallax*.

The *background* is defined as the static part in the motion-compensated sequence, whereas foreground objects are parts that has non-zero residual motion, either due to relative motion wrt the camera or to parallax wrt the stabilization plane.

The definition of background implies that it can be recovered as a *mosaic*, as the absence of residual motion is what allows to seamlessly compose images together (after a suitable transformation) into a larger aggregate.

It is well known that, if i) the scene is planar or ii) the point of view does not change (pure rotation), two pictures of the same static scene are related by a non-singular linear transformation of the projective plane (or *homography*).

Inter-frame homography computation is based on correspondences for details produced by the Kanade-Lucas-Tomasi (KLT) tracker [27], initialized with phase-correlation to reduce search range. Assuming that the majority of the tracked features belong to the background, Least Median of Squares is used to be robust against tracking errors and features attached to moving objects. Finally, given the set of *inlier* point matches, the homography is computed according to a technique proposed in [14], which obtains an optimal estimate and reduces the instability of images alignment even with a small overlap between frames. A frame is chosen as the reference one, then, for

[1] A video shot is defined as an image sequence captured with a single operation of the camera and presenting a continuous action in time and space [1].

Figure 1. Ego-motion compensation. Frames are warped so as to compensate for camera motion.

each other frame, the stabilizing homography is obtained by combining the inter-frame homographies. All the frames are warped accordingly to produce a new video sequence where the background is static (Fig. 1). More details on this technique are given in [18].

4. Background modeling

As we have discussed in the previous section, foreground objects can be extracted effectively by subtracting the background in the image frames, provided that the background can be estimated.

Consider the stabilized sequence: Starting from a single pixel in one frame, a temporal line (or *time-line*) piercing all the aligned frames will intersect pixels that correspond to the background and pixels belonging to foreground. Our method is based on the following hypothesis (as in [6]): (i) the background is stationary; (ii) along each time-line the background is revealed at least once.

The first hypothesis implies that the same background point is imaged always onto the same pixel. The second hypothesis implies that no object can occlude the background for the entire sequence. Please note that this is necessary as we want to use only *observed* values to fill the background at each location.

If hypothesis ii) were stronger, requiring that along each time-line the background is revealed for more than 50% of the entire sequence length, the background could be easily obtained as the median value along the time-line. The technique presented here can deal, in principle, with sequences where the background is revealed exactly once.

We model the stabilized video sequence as a 3D array $\mathbf{v}_{x,y,t}$ of pixel values. Each entry contains a color value, which is a triplet (R,G,B). A 3D *patch* $\mathbf{v}_{\mathcal{S}}$ is a sub-array of the video sequence, defined in terms of the ordered set of its pixel coordinates: $\mathcal{S} = I_x \times I_y \times I_t$, where I_x, I_y, I_t are set of indexes. The set $\mathcal{W} = I_x \times I_y$ is the *spatial footprint* of the patch. A 2D patch $\mathbf{v}_{\mathcal{R}}$ (or *image patch*) is a 3D patch with a singleton temporal index: $\mathcal{R} = \mathcal{W} \times \{t\}$ or $\mathcal{R} = (\mathcal{W}, t)$.

4.1. Estimating image noise.

The first step is to estimate the noise affecting pixel values in the video sequence. In the following we shall assume that the three color channels (R,G,B) are statistically independent, therefore we will consider here one color channel at a time.

Assuming that noise is i.i.d. Gaussian with zero-mean $\mathcal{N}(0, \sigma_m^2)$, the pixel values of the video sequence of length $L-1$ obtained by subtracting each consecutive frame: $\mathbf{n}_{x,y,t} = \mathbf{v}_{x,y,t} - \mathbf{v}_{x,y,t+1}$ are distributed with $\mathcal{N}(0, 2\sigma_m^2)$ plus outliers due to foreground objects. The noise standard deviation σ_m is then estimated robustly from $\mathbf{n}_{x,y,t}$. In order to get more statistics, we consider not only the difference between consecutive frames but also frames of distance two and three.

A robust estimator of the spread of a distribution is given by the Median Absolute Difference (MAD):

$$\text{MAD} = \text{med}_i\{|\mathbf{n}_i - \text{med}_i\{\mathbf{n}_i\}|\}. \quad (1)$$

It can be seen [7] that, for symmetric distributions, the MAD coincides with the inter-quartile range, hence, in our case:

$$\text{MAD} = \frac{1}{2}\left(\Phi^{-1}(\tfrac{3}{4}) - \Phi^{-1}(\tfrac{1}{4})\right)\sqrt{2}\sigma_m = \Phi^{-1}(\tfrac{3}{4})\sqrt{2}\sigma_m \approx 0.9539\sigma_m. \quad (2)$$

where $\Phi^{-1}(\alpha)$ is the α-th quantile of the cumulative normal distribution.

4.2. Temporal clustering

The spatial indexes are subdivided into windows $\mathcal{W}_i$ of size $N \times N$, overlapping by half of their size in both dimensions as shown in Fig. 3.

Let $\mathbf{v}_{\mathcal{S}}$, $\mathcal{S} = \mathcal{W}_i \times \{1 \cdots L\}$, be a patch of footprint $\mathcal{W}_i$ which extends in time from the first to the last frame. In order to reduce temporal redundancy, in each 3D patch $\mathbf{v}_{\mathcal{S}}$ we cluster image patches that depict the same static portion of the scene with *single linkage agglomerative clustering* [11]. Starting from all singletons, each sweep combines two clusters into a single cluster. After establishing a distance between objects, a method needs to be chosen to determine

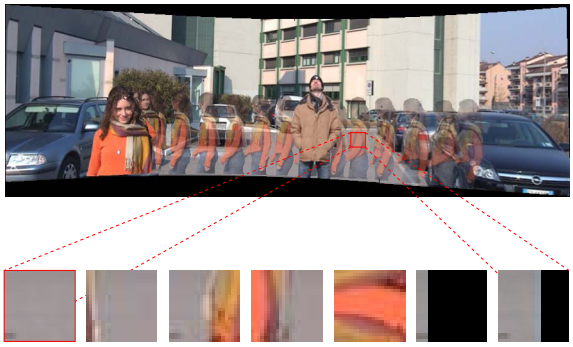

Figure 2. Clustering example. The top image summarizes the motion compensated "Dado" sequence and the images in the bottom row are (magnified) cluster representatives insisting on the highlighted footprint.

which two groups should be linked. The *simple linkage* rule says that the two groups that achieve the smallest inter-group distance between any pair of objects are linked. A *cutoff distance*, i.e., a distance behind which two clusters are not linked, is to be set.

In our case, the distance between two image patches $\mathbf{v}_{(\mathcal{W},t_1)}$ and $\mathbf{v}_{(\mathcal{W},t_2)}$ is given by the Sum of Squared Distances (SSD):

$$\mathrm{SSD}(\mathcal{W},t_1,t_2) = \frac{1}{2\sigma_m^2}\sum_{x,y\in\mathcal{W}} ||\mathbf{v}_{x,y,t_1} - \mathbf{v}_{x,y,t_2}||^2 \quad (3)$$

The cutoff distance should prevent clustering together image patches that do not depict the same objects. It is obtained from a statistical test, based on the expected distribution of the SSD between two image patches that depict the same *static* portion of the scene. The SSD has a Chi-square distribution with $3N^2$ degrees of freedom, which is evident if we re-write (3) as a *Mahalanobis* distance:

$$\mathrm{SSD}(\mathcal{W},t_1,t_2) = (\bar{\mathbf{v}}_{\mathcal{W},t_1} - \bar{\mathbf{v}}_{\mathcal{W},t_2})^\top (2\sigma_m^2\mathbf{I})^{-1} (\bar{\mathbf{v}}_{\mathcal{W},t_1} - \bar{\mathbf{v}}_{\mathcal{W},t_2}) \quad (4)$$

where $\bar{\mathbf{v}}_{\mathcal{W},t}$ is the $3N^2$-dimensional vector obtained by "vectorizing" $\mathbf{v}_{\mathcal{W},t}$ (because $N^2 = |\mathcal{W}|$, and 3 is the number of color channels).

Therefore, given a desired confidence level α, we deem that image patches $\mathbf{v}_{\mathcal{W},t_1}$ and $\mathbf{v}_{\mathcal{W},t_2}$ depict the same static portion of the scene (hence they can be linked in the clustering) if:

$$\mathrm{SSD}(\mathcal{W},t_1,t_2) < \chi^{-1}_{3N^2}(\alpha) \quad (5)$$

where $\chi_n^{-1}(\alpha)$ is α-th quantile of the cumulative Chi-square distribution with n d.o.f.

Although clusters are made of image patches instead of pixels, the clustering phase implements the same idea as the *intervals of stable intensity* defined in [17], except for clusters do not need to form a connected temporal interval, and there are no fancy thresholds.

The resulting clusters are 3D patches, with possibly not consecutive temporal indexes. Let $\mathcal{W}\times\mathcal{T}_k$ denote cluster k over spatial footprint $\mathcal{W}$, a representative image patch for that cluster is obtained by averaging pixel values along the time-line:

$$\mathbf{u}_{x,y,k} = \frac{1}{|\mathcal{T}_k|}\sum_{t\in\mathcal{T}_k}\mathbf{v}_{x,y,t} \quad \forall x,y\in\mathcal{W}. \quad (6)$$

As a consequence, the noise affecting the values $\mathbf{u}_{x,y,k}$ is i.i.d. $\mathcal{N}(0,\sigma_k^2)$ with $\sigma_k^2 = \frac{\sigma_m^2}{|\mathcal{T}_k|}$.

In each spatial footprint $\mathcal{W}$ we have now a variable number of cluster representatives $\mathbf{u}_{x,y,k_1}\ldots\mathbf{u}_{x,y,k_\ell}$ (see Fig. 2). The underlying assumption is that (at least) one of them depicts *only* static background: The subsequent stage is devoted to find out which one.

A heuristic that demonstrated helpful to cull the clusters is discarding clusters composed by only one frame provided that this do not eliminate *all* the clusters insisting on a footprint. This is related to the practice of discarding patches with high *motion energy* [22, 6]. In our case, as the SSD is related to the motion energy, image patches with high motion energy tends to form cluster of size one.

By introducing this heuristic, we implicitly strengthen our hypothesis, requiring that along each time-line the background is revealed at least *twice*.

4.3. Background tessellation

The background is constructed with a sequential approach: Starting from seed patches, a tessellation is grown by choosing, at each site, the best continuation of the current background.

The background seeds are the representatives of the largest clusters. Since we assume that no foreground object is stationary in *all* the frames, if the largest clusters have size L (maximal), the seeds are fully reliable. Otherwise, mistakes are possible.

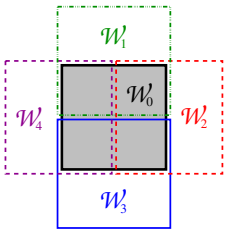

Figure 3. Overlapping footprints.

Figure 4. Snapshots of the background as the tessellation proceeds.

The growing proceeds as follows. Let $\mathcal{W}_0$ be a spatial footprint where a background patch has already been assigned. We consider in turn each of the four footprints that overlap with it: $\mathcal{W}_i, i = 1, \ldots, 4$, (see Fig. 3), and try to assign a background to each of them (if it was not already assigned) by choosing one from the cluster representatives that insist on $\mathcal{W}_i$. The selected patch has to fulfill two requirements:

i) in the part that overlaps with $\mathcal{W}_0$ it has to depict the same scene points as the background patch, so that it can be stitched seamlessly to it;

ii) in the non-overlapping part it has to represent the "best continuation" of the background.

This procedure is repeated for all the footprints, until all the background has been assigned (Fig. 4).

As for the first requirement, the discrepancy of a candidate image patch $\mathbf{u}_{(\mathcal{W}_i,k)}$ with the background patch $\mathbf{u}_{(\mathcal{W}_0,k_0)}$ in the overlapping part is measured with:

$$\mathrm{SSD}(\mathcal{W}_0 \cap \mathcal{W}_i, k_0, k) = \frac{1}{\sigma_{k_0}^2 + \sigma_k^2} \sum_{x,y \in \mathcal{W}_0 \cap \mathcal{W}_i} ||\mathbf{u}_{x,y,k_0} - \mathbf{u}_{x,y,k}||^2. \quad (7)$$

By the same token as before (Eq. (5)), $\mathbf{u}_{(\mathcal{W}_i,k)}$ is considered for inclusion in the background with confidence α if

$$\mathrm{SSD}(\mathcal{W}_0 \cap \mathcal{W}_i, k_0, k) < \chi_{3M}^{-1}(\alpha) \quad (8)$$

If $\mathcal{W}_i$ happens to overlap with other footprints than $\mathcal{W}_0$ where the background has already been assigned, the same test is applied, *mutatis mutandi*, to the entire area of overlap.

As for the second requirement, we propose here a method to compare two candidates (if there are more candidates a round robin tournament is used), based on the principles of *visual grouping* [28]. The approach rest on the observation that foreground objects generally introduce a discontinuity with the background (as in [9]). When a pure background patch is compared to an image patch containing foreground, their binarized difference defines a partitioning of the pixels into two groups (Fig. 5), i.e., a segmentation. The previous observation implies that the score of this segmentation according to the principles of visual grouping (similarity, proximity, and good continuation) must be higher in the patch containing foreground than in the one containing background. This links the problem of selecting

Figure 5. From left to right: two cluster representatives candidate to fill a background patch and their binarized difference.

the best continuation of the background to the visual grouping theory.

Graphs cuts have been proposed in [25] as general computational framework for grouping. The image is represented as a complete weighted undirected graph $G = (V, E)$, by taking each pixel as a node and connecting each pair of pixels by an edge. The weight on that edge reflects the likelihood that the two pixels belong to the same region. Grouping is cast as the problem of partitioning the vertices into disjoint sets, where the similarity among the vertices in a set is high and across different sets is low. The edge weight connecting the two nodes i and j is defined as [25]:

$$w_{ij} = e^{-(\mathbf{f}_i - \mathbf{f}_j)^\top (2\mathbf{\Lambda})^{-1} (\mathbf{f}_i - \mathbf{f}_j)} \quad (9)$$

where $\mathbf{f}_i$ is a feature vector containing the spatial position of a pixel i, x_i and y_i, and its RGB color values, R_i, G_i, B_i: $\mathbf{f}_i = [x_i, y_i, R_i, G_i, B_i]$. The diagonal matrix $\mathbf{\Lambda}$ contains normalizing values, which are approximately (the square of) 1/4 of the range of variability of the respective component: $\mathbf{\Lambda}^{1/2} = \mathrm{diag}(N/4, N/4, \sigma_m, \sigma_m, \sigma_m)$.

The graph can be partitioned into two disjoint sets, A and B, $A \cup B = V$, $A \cap B = \emptyset$, by simply removing edges connecting the two parts. This set of edges constitute a *cut*. The cost of the cut, which measures the degree of similarity between the two region A and B, is the sum of all its edge weights:

$$cut(A, B) = \sum_{i \in A, j \in B} w(i, j) \quad (10)$$

The optimal segmentation is the cut with the minimal cost.

Going back to the problem of choosing between two image patches the one that yields the best continuation of the background, consider the cut defined by their binarized difference:

$$A = \{(x, y) : (\mathbf{u}_{x,y,k_1} - \mathbf{u}_{x,y,k_2})^\top (\sigma_{k_1}^2 \mathbf{I} + \sigma_{k_2}^2 \mathbf{I})^{-1} (\mathbf{u}_{x,y,k_1} - \mathbf{u}_{x,y,k_2}) < \chi_3^{-1}(\alpha)\} \quad (11)$$

The patch where $cut(A, B)$ is lower, is the one containing the foreground pixels (because the cut is along the discontinuity), whereas the same cut in the background patch has a higher cost, because – not being correlated with the structure of the background patch – it is more likely to contain expensive edges.

Our method based on graph-cuts can be seen as a principled way of applying the same continuity criterion as in [9], where a heuristic based on the comparison of the *inner* and *outer* boundaries of the difference region is employed.

5. Foreground segmentation

As the footprints are overlapping, on a single pixel (x, y) in the final background image might insist up to four patches. Let $\mathcal{T}$ be the set of temporal indexes of the frames that contributed to the background value at (x, y), via the cluster representatives. The estimate of the background color $\mathbf{c}_{x,y}$ and its variance $\sigma^2_{x,y}$ are obtained as the sample mean and variance – respectively – of the values $\mathbf{v}_{x,y,\mathcal{T}}$. A sample variance image is shown in Fig. 6. Due to small misalignment errors and to the low pass effect introduced by warping, the edges have a higher variance.

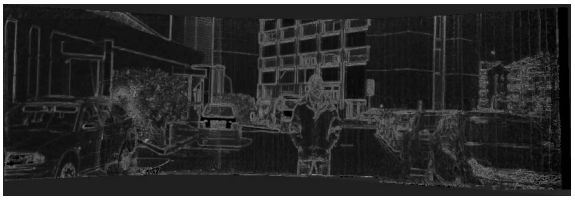

Figure 6. Gray level visualization of the per-pixel variance of the "Dado" background (values are normalized in [0,255]).

Foreground/background segmentation is cast as the problem of testing (at a desired confidence level) whether two values comes from the same Gaussian distribution. Using again the Mahalanobis distance, a pixel $\mathbf{v}_{x,y,t}$ of the stabilized sequence is deemed to belong to the background with confidence α if:

$$(\mathbf{v}_{x,y,t} - \mathbf{c}_{x,y})^\top (\sigma^2_m \mathbf{I} + \sigma^2_{x,y} \mathbf{I})^{-1} (\mathbf{v}_{x,y,t} - \mathbf{c}_{x,y}) < \chi_3^{-1}(\alpha)$$

This defines a binary image that "masks" foreground objects, which is then cleaned with morphological filtering. Examples of binary masks are shown in Fig. 7.

6. Results

In this section, we report some results obtained by applying our technique to video shots acquired with a digital hand-held camera. The sequences were selected to set different challenges to our algorithm. Fig. 8 shows the results organized in a table, one row for each sequence.

Figure 7. Binary masks obtained after segmenting frames shown in Fig. 1.

In the experiments we used the following parameter setting: $N = 17$ and a confidence level $\alpha = 0.999999$.

In the "Dado" sequence, a young woman is walking from right to left and passes behind a man who is standing still; the camera does a panning motion following the walking woman. In order to clutter the background, the woman stops for a while and then walks towards the camera. Indeed, she is still visible in the median image, whereas our method recovers the clean background. Despite the young woman is fragmented in several parts, before and after the occlusion, our tracking phase can recover from over-fragmentation and recognizes it as a single VO.

The "Road-sign" sequence depicts a road-sign in front of a building. As the background (the facade) is planar, the camera can move freely. In the motion-compensated sequence, the road-sign moves due to parallax. Its motion, however, is not sufficiently prominent to make it disappear in the median image. Our method, instead, first recovers the background without occlusions and then extract the road-sign as an independent Video Object.

In the "Yard" sequence, two persons that are standing still for more than half of the sequence, start to walk towards each other and then cross in the middle. The camera does a panning motion, following the young woman from right to left. The clutter is severe also in this case, indeed the median fails to obtain a clean background whereas our background modeling method succeeded. Despite the two persons overlap in the image for a significant number of frames, our technique is able to track them trough the video shot.

Original sequences and results are available on the web (http://profs.sci.univr.it/˜fusiello/demo/bkg).

7. Conclusions

We illustrated a method for video objects segmentation in a video sequence based on background recovery. The method is robust, as it can cope with serious occlusions caused by moving objects. It is scalable, as it can deal with any number of frames greater or equal than two. It is effective, as it always recovers the background when the

Figure 8. For each sequence, the top row shows some selected frames, the middle row shows the mosaic of the motion-compensated background obtained using the median operator (left) and our technique (right), and the bottom row shows some selected frames of the Video Object sequence.

assumptions are satisfied. Moreover, our method rests on sound principles in all its stages, and only few, intelligible parameters are needed, namely the confidence level for the tests and the patch size. Future work will aim at estimating it from the data, using a multi-resolution approach. We also plan to include a shadow removing stage, as shadows can deceive foreground segmentation [24].

References

[1] R. Brunelli, O. Mich, and C. M. Modena. A survey on the automatic indexing of video data. *Journal of Visual Communication and Image Representation*, 10:78–112, 1999.

[2] A. Colombari, M. Cristani, V. Murino, and A. Fusiello. Exemplar-based Background Model Initialization. In *Proceedings of the 3rd ACM International Workshop on Video Surveillance & Sensor Networks*, 2005.

[3] A. Colombari, A. Fusiello, and V. Murino. Background initialization in cluttered sequences. In *5th Workshop on Perceptual Organization in Computer Vision*, 2006. In conjunction with CVPR 2006.

[4] A. Colombari, A. Fusiello, and V. Murino. Segmentation and tracking of multiple video objects. *Pattern Recognition*, 40(4):1307–1317, April 2007.

[5] P. Giaccone and G. Jones. Segmentation of global motion using temporal probabilistic classification. In *British Machine Vision Conference*, pages 619–628, 1998.

[6] D. Gutchess, M. Trajkovic, E. Cohen-Solal, D. Lyons, and A. Jain. A background model initialization algorithm for video surveillance. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 733–740, 2001.

[7] F. Hampel, P. Rousseeuw, E. Ronchetti, and W. Stahel. *Robust Statistics: the Approach Based on Influence Functions*. Wiley Series in probability and mathematical statistics. John Wiley & Sons, 1986.

[8] I. Haritaoglu, D. Harwood, and L. Davis. W^4: Who? When? Where? What? a real time system for detecting and tracking people. In *Proceedings of the 3rd International Conference on Face and Gesture Recognition*, 1998.

[9] C. Herley. Automatic occlusion removal from minimum number of images. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 1046–1049, 2005.

[10] M. Irani, P. Anandan, J. Bergen, R. Kumar, and S. Hsu. Efficient representations of video sequences and their applications. *Signal processing: Image Communication*, 8(4):327–351, May 1996.

[11] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.

[12] O. Javed, K. Shafique, and M. Shah. Hierarchical approach to robust background subtraction using color and gradient information. In *Workshop on Motion and Video Computing*, pages 22–27, 2002.

[13] J. Jia, T. Wu, Y. Tai, and C. Tang. Video repairing: Inference of foreground and background under severe occlusion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[14] K. Kanatani and N. Ohta. Accuracy bounds and optimal computation of homography for image mosaicing applications. In *International Conference on Computer Vision*, volume 1, pages 73–79, September 1999.

[15] R. Koenen, F. Pereira, and L. Chiariglione. MPEG-4: Context and objectives. *Signal Processing: Image Communications*, 9(4):295–304, 1997.

[16] L. Li and M. K. H. Leung. Integrating intensity and texture differences for robust change detection. *IEEE Transactions on Image Processing*, 11(2):105–112, February 2002.

[17] W. Long and Y. Yang. Stationary background generation: An alternative to the difference of two images. *Pattern Recognition*, 23:1351–1359, 1990.

[18] F. Odone, A. Fusiello, and E. Trucco. Layered representation of a video shot with mosaicing. *Pattern Analysis and Applications*, 5(3):296–305, August 2002.

[19] N. Paragios and V. Ramesh. A mrf-based approach for real-time subway monitoring paragios. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1034–1040, 2001.

[20] K. A. Patwardhan, G. Sapiro, and M. Bertalmio. Video inpainting of occluding and occluded objects. In *Proceedings of the IEEE International Conference on Image Processing*, volume 2, pages 69–72, 2005.

[21] P.Nunes, P.Correia, and F.Pereira. Coding video objects with the emerging mpeg-4 standard. In *I Conferência Nacional de Telecomunicações*, April 1997.

[22] C. Rasmussen and T. Korah. Spatiotemporal inpainting for recovering texture maps of partially occluded building facades. In *Proceedings of the IEEE International Conference on Image Processing*, volume 3, pages 125–128, 2005.

[23] P. L. Rosin. Thresholding for change detection. In *Proceedings of the Sixth International Conference on Computer Vision*, page 274. IEEE Computer Society, 1998.

[24] E. Salvador, A. Cavallaro, and T. Ebrahimi. Cast shadow segmentation using invariant color features. *Comput. Vis. Image Underst.*, 95(2):238–259, 2004.

[25] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[26] C. Stauffer and W. E. L. Grimson. Learning patterns of activity using real-time tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.

[27] C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, Pittsburg, PA, April 1991.

[28] M. Wertheimer. Laws of Organization in Perceptual Forms. In Ellis Willis D., editor, *A Source Book of Gestalt Psychology*, pages 71–88. Harcourt Brace, New York, 1939.

[29] Y. Wexler, E. Shechtman, and M. Irani. Space-time video completion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 120–127, 2004.

[30] C. Wren, A. Azarbayehani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:780–785, 1997.

[31] D. S. Zhang and G. Lu. Segmentation of moving objects in image sequence: A review. *Circuits, Systems and Signal Processing*, 20(2):143–183, 2001.