A Probabilistic Framework for Spatio-Temporal Video Representation & Indexing

Hayit Greenspan[1], Jacob Goldberger[2], and Arnaldo Mayer[1]

[1] Faculty of Engineering, Tel Aviv University, Tel Aviv 69978, Israel
[2] CUTe Ltd., Tel-Aviv, Israel

Abstract. In this work we describe a novel statistical video representation and modeling scheme. Video representation schemes are needed to enable segmenting a video stream into meaningful video-objects, useful for later indexing and retrieval applications. In the proposed methodology, unsupervised clustering via Guassian mixture modeling extracts coherent space-time regions in feature space, and corresponding coherent segments (*video-regions*) in the video content. A key feature of the system is the analysis of video input as a single entity as opposed to a sequence of separate frames. Space and time are treated uniformly. The extracted space-time regions allow for the detection and recognition of video events. Results of segmenting video content into static vs. dynamic video regions and video content editing are presented.

1 Introduction

Video search in large archives is a growing research area. Advanced video representation schemes are needed to provide for compact video storage as well as a concise model for indexing and retrieval applications. Segmenting an input video stream into interesting "events" is becoming an important research objective. The goal is to progress towards content-based functionalities, such as search and manipulation of objects, semantic description of scenes (e.g.,"indoor" vs. "outdoor"), detection of unusual events and recognition of objects. This work focuses on video data, video representation and segmentation.

As a first step in organizing video data, a given video clip is parsed in the temporal domain into short video shots, each of which contains consistent visual content. A video shot can be considered as a basic unit of video data. Since visual information is similar in each shot, global image features such as color, texture and motion can be extracted and used for the search and retrieval of similar video shots.

In order to further exploit the video content, a video shot needs to be decomposed into meaningful *regions* and *objects*, so that search, retrieval and content manipulation based on object characteristics, activities, and relationships are possible. Video *indexing* is concerned with segmenting the video stream into

A. Heyden et al. (Eds.): ECCV 2002, LNCS 2353, pp. 461–475, 2002.
© Springer-Verlag Berlin Heidelberg 2002

meaningful video-objects that may be useful for later indexing and retrieval applications.

Video has both spatial and temporal dimensions and hence a good video index should capture the spatio-temporal contents of the scene. Video objects (otherwise termed, "space-time objects" [3], "subobjects" [7]) are generally extracted via a two-stage processing framework consisting of frame-by-frame spatial segmentation followed by temporal tracking of information across frames. In the presented work a novel statistical framework is proposed for modeling and segmenting video content into coherent space-time segments within the video frames and across frames. We term such segments "video-regions". Unsupervised clustering, via Gaussian mixture modeling (GMM), enables the extraction of space-time clusters, or "blobs", in the representation space, and the extraction of corresponding video-regions in the segmentation of the video content. An important differentiation from existing work is that the video is modeled as a single entity, as opposed to a sequence of separate frames. Space and time are treated uniformly in a single-stage modeling framework.

The paper is organized as follows. Section 2 describes related work in the literature. Section 3 focuses on the representation phase of the proposed framework in which we transition from pixels to coherent regions in feature space, via Gaussian mixture modeling. A space-time video representation model is described. Section 4 discusses the extension to probabilistic video segmentation. The probabilistic representation enables the definition and detection of events in video. Experimental results of video event detection are presented in section 5. A discussion concludes the paper in Section 6.

2 Previous Work on Video Representation and Segmentation

Research in content-based video retrieval focused initially on ways of searching video clips based on global similarities, such as color, texture, and motion (e.g., [6], [10], [12], [14], [20], [21], [15], [19]). A separate set of works has started to address localized, regional representations that enable spatio-temporal segmentation for object-based video retrieval (e.g., [7], [3], [2], [16]).

Spatio-temporal segmentation has been a very challenging research problem, and many algorithms are proposed in the literature ([8], [3], [12], [17]). Many approaches use optical flow methods (e.g., [11], [13]) to estimate motion vectors at the pixel level, and then cluster pixels into regions of coherent motion to obtain segmentation results. Due to the complexity of object motion in general videos, pure motion-based algorithms cannot be used to automatically segment and track regions through image sequences. The drawbacks include the fact that optical flow does not cope well with large motion, and the fact that regions of coherent motion may contain multiple objects and need further segmentation for object extraction.

In works that incorporate spatial segmentation into motion segmentation, it is commonly the case that the spatio-temporal segmentation task is decomposed into two separate tasks of spatial segmentation (based on in-plane features such as color and texture) within each frame in the sequence, or within a selected frame of the sequence, followed by a motion segmentation phase. In [3] color and edge features are used to segment a frame into regions. Optical flow, computed for each pair of frames, is utilized to project and track color regions through a video sequence. Given color regions and the generated optical flow, a linear regression algorithm is used to estimate the affine motion for each region. In [7] a six-parameter two-dimensional (2-D) affine transformation is assumed for each region in the frame, and is estimated by finding the best match in the next frame. Multiple objects with the same motion are separated by spatial segmentation.

The challenge in video indexing is to utilize the representation model and the segmentation ability towards the definition of meaningful *regions* and *objects* for future content analysis. The shift to regions and objects is commonly accomplished by two-phase processing: a segmentation process followed by the tracking of regions across segmented frames. In [3] a region is defined as a contiguous set of pixels that is homogeneous in the features that we are interested in (such as color, texture, shape). A video object is then defined as a collection of video regions that have been grouped together under some criteria across several frames. Namely, a video object is a collection of regions exhibiting consistency across several frames in at least one feature. A hierarchical description of video content is discussed in [7]. A video shot is decomposed into a set of sub-objects. The sub-objects consist of a sequence of tracked regions, with the regions obtained by segmentation.

This paper belongs to the family of works (as cited above) that propose a regional video content description, and provide a means for finding information in the video without any high-level understanding of the actual content. The focus is on extracting video-regions as coherent space-time video segments. A key feature of the current work is the use of a *statistical* methodology for describing the video content. The video content is modeled in a continuous and probabilistic space. No geometrical modeling constraints (e.g., planarity), or object rigidity constraints need to be imposed as part of the motion analysis. No separate segmentation and motion-based tracking schemes are used.

3 Learning a Probabilistic Model in Space-Time

The modeling phase is accomplished in the feature space. A transition is made from the pixel representation to a mid-level representation of an image, in which the image is represented as a set of coherent regions in feature space. Unsupervised clustering using a Gaussian mixture model (GMM) is pursued to form meaningful groupings in feature space and a corresponding localized space-time representation in the image plane.

3.1 Feature Extraction

In this work we use the color and relative color layout, as the main characteristics of the representation. *Color features* are extracted by representing each pixel with a three-dimensional color descriptor in the $L * a * b$ color space, which was shown to be approximately perceptually uniform; thus distances in this space are meaningful [18]. In order to include *spatial information*, the (x, y) position of the pixel is appended to the feature vector. Including the position generally decreases oversegmentation and leads to smoother regions. The *time feature* (t) is added next. The time descriptor is taken as an incremental counter: $1, \ldots,$ number of frames in shot. Each of the features is normalized to have a value between 0 and 1.

Following the feature extraction stage, each pixel is represented with a six-dimensional feature vector, and the image-sequence as a whole is represented by a collection of feature vectors in the six-dimensional space. Note that the dimensionality of the feature vectors, and the feature space, is dependent on the features chosen and may be augmented if additional features are added.

3.2 Grouping in the Space-Time Domain

In this stage, pixels are grouped into homogeneous regions, by grouping the feature vectors in the selected six-dimensional feature space. The underlying assumption is that the image colors and their space-time distribution are generated by a mixture of Gaussians. The feature space is searched for dominant clusters and the image samples in feature space are then represented via the modeled clusters. Note that although image pixels are placed on a regular (uniform) grid, this fact is not relevant to the probabilistic clustering model in which the affiliation of a pixel to the model clusters is of interest. In general, a pixel is more likely to belong to a certain cluster if it is located near the cluster centroid. This observation implies a unimodal (Gaussian) distribution of pixel positions within a cluster. Each homogeneous region in the image plane is thus represented by a Gaussian distribution, and the set of regions in the image is represented by a Gaussian mixture model. Learning a Gaussian mixture model is in essence an unsupervised clustering task.

The Expectation-Maximization (EM) algorithm is used [5], to determine the maximum likelihood parameters of a mixture of k Gaussians in the feature space. The image is then modeled as a Gaussian mixture distribution in feature space. We briefly describe next the basic steps of the EM algorithm for the case of Gaussian mixture model. The distribution of a random variable $X \in R^d$ is a mixture of k Gaussians if its density function is:

$$f(x|\theta) = \sum_{j=1}^{k} \alpha_j \frac{1}{\sqrt{(2\pi)^d |\Sigma_j|}} \exp\{-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)\} \qquad (1)$$

such that the parameter set $\theta = \{\alpha_j, \mu_j, \Sigma_j\}_{j=1}^{k}$ consists of:

- $\alpha_j > 0 \quad , \quad \sum_{j=1}^{k} \alpha_j = 1$
- $\mu_j \in R^d$ and Σ_j is a $d \times d$ positive definite matrix.

Given a set of feature vectors $x_1, ..., x_n$, the maximum likelihood estimation of θ is :

$$\theta_{ML} = \arg\max_{\theta} f(x_1, ..., x_n|\theta) \tag{2}$$

The EM algorithm is an iterative method to obtain θ_{ML}. Given the current estimation of the parameter set θ, The first step in applying the EM algorithm to the problem at hand is to initialize the mixture model parameters. The K-means algorithm [9] is utilized to extract the data-driven initialization. The iterative updating process is repeated until the log-likelihood is increased by less than a predefined threshold from one iteration to the next. In this work we choose to converge based on the log-likelihood measure and we use a 1% threshold. Other possible convergence options include using a fixed number of iterations of the EM algorithm, or defining target measures, as well as using more strict convergence thresholds. We have found experimentally that the above convergence methodology works well for our purposes. Using EM, the parameters representing the Gaussian mixture are found.

An optimality criterion for k is based on a tradeoff between performance and number of parameters used for describing the mixture distribution. The Minimum Description Length (MDL) [4] is such a criterion that has been used for selecting among values of k in still image processing [1]. Using the MDL principle, the K-Means and EM are calculated for a range of k values, $k \geq 1$, with k corresponding to the model size. The model for which the MDL criterion is maximized is chosen. When models using two values of k fit the data equally well, the simpler model will be chosen.

3.3 Model Visualization

For model visualization purposes we start with showing a still image example, Figure 1. The representation in the space-time domain is shown, for a single blob, in Figure 2. In Figure 1, the GMM model is learned for a given static image in a five-dimensional feature space (color and spatial features). The input image is shown (top) and a set of localized Gaussians representing the image for differing mixtures (different k values), bottom. In this visualization each localized Gaussian mixture is shown as a set of ellipsoids. Each ellipsoid represents the support, mean color and spatial layout, of a particular Gaussian in the image plane.

The transition to the space-time domain is more difficult to visualize. Figure 2 shows two scenarios of a particular blob from within a GMM in the space-time domain (the shown blob represents a car in varying segments of the video sequence shown in Figure 5). In this case we use a three-dimensional space to represent the ellipsoid support and spatial layout in space-time. The mean color

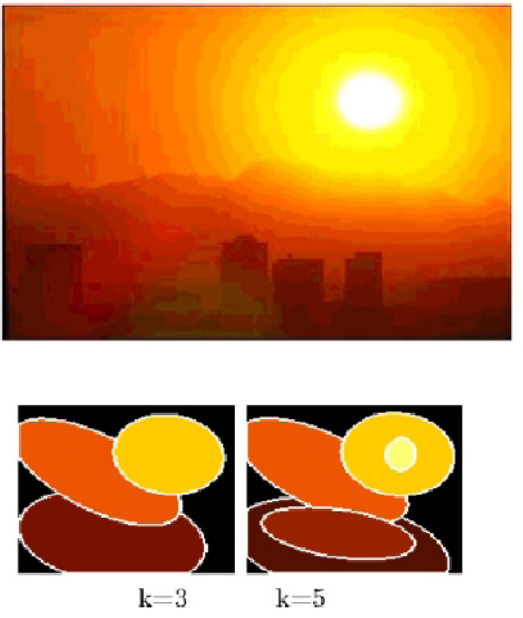

Fig. 1. Example of a still input image (top) with the corresponding set of representative Gaussian mixtures (bottom). The mixtures are composed of $k = 3$ and 5 components. Each ellipsoid represents the support, mean color and spatial layout, of a particular Gaussian in the image plane.

of the ellipsoid indicates the region color characteristics. Planes are superimposed to show the connection between the six-dimensional space-time space and the five-dimensional image space without the time feature. The planes are positioned at specific frame time-slots. Projection of the six-dimensional blob onto a plane corresponds to a reduced model, in the image plane, similar to the example shown in Figure 1. Space-time characteristics of each Gaussian (blob) can be extracted from the generated model, in particular, from the covariance matrix of each Gaussian. These characteristics are evident within the visualization scheme. A blob representing a static video-region is shown in Figure 2(a). Note that no correlation is evident between the space (x, y) and the time (t) axis. A blob representing a dynamic video-region is shown in Figure 2(b). In this case, a strong positive correlation exists in the x and t dimensions. The projection of the blob onto the planes positioned at differing time intervals (or frame numbers) demonstrates the shifts of the blob cross-sections in the x, t direction. As t increases (corresponding to increasing frame number in the video sequence), the spatial support shift horizontally. Such a correlation indicates horizontal movement. Thus, the model extracted indicates space-time characteristics such as the differentiation between static and moving blobs. The GMM generated for a given video sequence can be visualized as a set of such elongated ellipsoids ("bubbles")

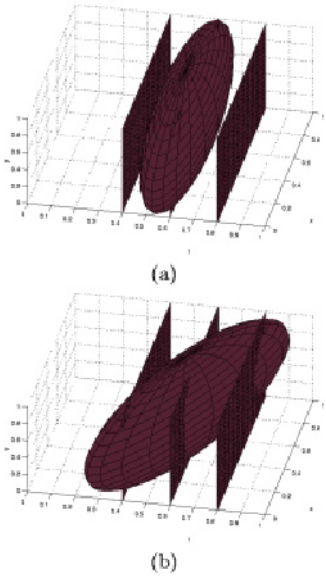

Fig. 2. A space-time elongated blob from a GMM in 6-dimensional feature space. The ellipsoid support and spatial layout in space-time is presented in 3D. The mean color of the ellipsoid indicates the region color characteristics. (a) a blob representing a static video-region. (b) a blob representing a dynamic video-region. In this case, a strong positive correlation exists in the x and t dimensions. Such a correlation indicates horizontal movement. Planes are superimposed to demonstrate the shifts of the blob cross-sections in the x, t direction.

within the three-dimensional space-time domain. The characteristics of the individual Gaussians within the model can be used for detection and recognition of video events, as will be further on explored in Section 5.

4 Probabilistic Video Segmentation

The six-dimensional GMM of color, space and time represents coherent regions in the combined space-time domain. A correspondence is now made between the coherent regions in feature space and localized temporally connected regions in the image plane of an individual frame and across frames in the video sequence. We segment the video sequence by assigning each pixel of each frame to the most probable Gaussian cluster, i.e. to the component of the model that maximizes the a-posteriori probability, as shown next.

The labeling of each pixel is done in the following manner: Suppose that the parameter set that was trained for the image is $\theta = \{\alpha_j, \mu_j, \Sigma_j\}_{j=1}^{k}$ Denote:

$$f_j(x|\alpha_j, \mu_j, \Sigma_j) = \tag{3}$$

$$\alpha_j \frac{1}{\sqrt{(2\pi)^d |\Sigma_j|}} \exp\{-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)\}$$

Equation (3) provides a probabilistic representation for the affiliation of each input sample, x, to the Gaussian components that comprise the learned model. The probabilistic representation is complete, in that no information is lost.

It is often desired to proceed with a decision phase that is based on the extracted probabilities and provides a "hard-decision" map of pixel affiliations into the predefined categories. The labeling of a pixel related to the feature vector x is chosen as the maximum a-posteriori probability, as follows:

$$\text{Label}(x) = \arg\max_j f_j(x|\alpha_j, \mu_j, \Sigma_j) \tag{4}$$

In addition to the labeling, a confidence measure can be computed. The confidence measure is a probabilistic label that indicates the uncertainty that exists in the labeling of the pixel. The probability that a pixel x is labeled j is:

$$p(\text{Label}(x) = j) = \frac{f_j(x|\alpha_j, \mu_j, \Sigma_j)}{f(x|\theta)} \tag{5}$$

with the denominator as defined in equation (1).

Equations (3-5) show that the video segmentation is probabilistic. For each frame, each sample feature vector (per pixel), x, is labeled and the label is projected down to the image plane. This method is applied frame by frame. A unique set of blobs is used for modeling the entire frame-sequence. Thus, the same blobs are used to segment each frame of the sequence. A by-product of the segmentation process is therefore the temporal tracking of individual frame regions. Each Gaussian or blob in feature space corresponds to a video-region. A video-region is linked to the properties of the corresponding blob.

An example of the GMM representation extracted per input frame sequence, along with the corresponding probabilistic segmentation results, can be seen in the top rows of Figure 5: row (a) presents a selection of input frames from the video sequence, row (b) presents a visualization of the space-time model, as related to the corresponding input frame, and row (c) shows the segmentation results projected down to each individual frame in the sequence. Each pixel from the original image is displayed with the color of the most-probable corresponding Gaussian. The segmentation results provide a visualization tool for better understanding the image model. Uniformly colored regions represent homogeneous regions in feature space. The associated pixels are all linked (unsupervised) to the corresponding Gaussian characteristics.

The EM algorithm ensures a Gaussian mixture in color, space and time. In essence, we have found the most dominant colors in the video sequence, as present in homogeneous localized regions in space-time, making up the video composition. Incorporating the spatial information into the feature vector does not only supply local information. It is also imposing a correlation between adjacent pixels such that pixels that are not far apart tend to be associated (labeled) with

the same Gaussian component. The segmentation results discussed above clearly demonstrate this fact, as can be seen in the smooth nature of the segmentation that results in labeling each individual frame according to the GMM.

5 Detection and Recognition of Events in Video

So far we have focused on the model generation process. In this section we investigate the model parameters further and show the connection between blobs and video events. A close look at the covariance matrix that represents each individual Gaussian blob in the Gaussian mixture model reveals several parameters that are space-time dependent. In Figure 3 we show a typical six-dimensional covariance matrix, along with three parameters of interest: $C_{t,x}, C_{t,y}$, and $C_{t,t}$.

We have defined a video-region as a particular sub-object segment in the video sequence that corresponds to a given Gaussian. Large values of $C_{t,x}$ indicate a strong correlation between the video-region horizontal position and time. In other words a horizontal movement of the region through the video sequence (note that horizontal and vertical directions refer to the x and y dimensions of the image plane, respectively). Similarly, $C_{t,y}$ reflects vertical movement. Small values of $C_{t,x}$ and $C_{t,y}$ suggest that the blob, and corresponding video-region is static. The time variance, $C_{t,t}$, represents the dispersion of the blob in the time domain (around the mean time coordinate, i.e. the time coordinate of the considered Gaussian' s center). A large $C_{t,t}$ value, for instance, indicates that the video-region is of extended duration, or is present in a majority of the frames that comprise the video sequence.

The correlation coefficient is defined as follows:

$$R_{i,j} = \frac{C_{i,j}}{\sqrt{C_{i,i}}\sqrt{C_{j,j}}}, \quad -1 \leq R_{i,j} \leq 1 \tag{6}$$

The range of the covariance parameters is bounded, thus enabling a comparison and thresholding process for detecting events of interest. The detection of *static vs. dynamic blobs* as well as the *magnitude of motion* in the image plane, is extracted via a thresholding process on the absolute values of $R_{t,x}$ and $R_{t,y}$. The *direction of motion* is extracted via the *sign* of $R_{t,x}$ and $R_{t,y}$.

The actual blob motion (pixels per frame) can be extracted using linear regression models in space and time, as shown in equation (7):

$$E(x|t = t_i) = E_x + \frac{C_{xt}}{C_{tt}}(t_i - E_t) \tag{7}$$

In this equation, horizontal velocity of the blob motion in the image plane is extracted as the ratio between the respective covariance parameters. Similar formalism allows for the modeling of any other linear motion in the image plane.

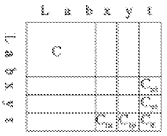

Fig. 3. A typical six-dimensional covariance matrix, along with three parameters of interest: $C_{t,x}, C_{t,y}$, and $C_{t,t}$.

5.1 Experimental Results

We show next a set of experiments in which we interact with video content, detect and recognize video events, all within the Gaussian mixture model representation framework. The sequences experimented with vary in length. In each example shown, a subset of the frames from the sequence is shown.

Blob editing. With the learned representation it is possible to edit blob characteristics and to project that to the image plane. The segmentation of the video sequence that follows the model generation, ensures that each blob is linked to pixels within a set of frames, and the pixels affiliated with a particular blob are linked to the blob characteristics. Any change in the blob characteristics will automatically be transmitted to the corresponding pixels within the respective frames in the video sequence. An equivalence may be seen with automatic annotation techniques in which a label is to be attached automatically to linked frames containing a particular object of interest.

We start with an experiment in which the goal is to perform *blob deletion*. In the specific scenario presented, the objective is to detect a moving car and replace the car-body region with static background. For this purpose we need to identify the blob associated with the desired video-region. An assumption for horizontal movement is used. The detection criterion therefore is based on the $R_{t,x}$ correlation coefficient. The correlation coefficient for the car blob (car-body region) is close to 1 while for the other blobs, that represent static background regions, it is an order of magnitude smaller. Once the moving blob is detected, the video sequence segmentation maps are used to generate a list of frames in which pixels are linked to the particular blob of interest. We term this list the "moving-blob frame list". A second list is made, "stationary-blob frame list", of frames that do not include pixels linked to the blob of interest. The blob deletion procedure involves replacing the pixels of the selected video region, in each frame belonging to the "moving-blob frame list", with pixels of same spatial coordinates extracted from a frame belonging to the "stationary-blob frame list".

Figure 5 shows a sequence of frames in which we see the original video data (a), followed in consecutive rows, with the sequence representation model (b),

segmentation maps (c), and final output of a new video sequence without the moving car-body region (d). In this experiment an input sequence of 8 frames was used. A 12-Gaussian mixture was used for the representation. In the output video sequence, the car-region has successfully been replaced with background information. The wheels have been preserved and thus remain as the only moving objects in the scene. A slight variation would lead to the wheels removal along with the car-body region.

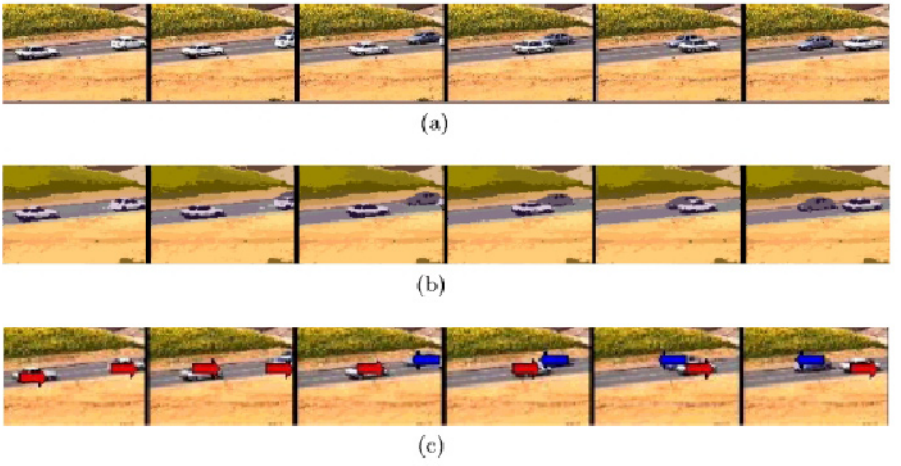

Fig. 4. Movement detection. The objective is to detect the cars as they enter the field-of-view and to distinguish left-to-right from right-to-left motion. A 21 frame sequence is used. The model size is 37 Gaussians. (a) The original video data; (b) The sequence segmentation maps; (c) Moving objects (dynamic blobs) are detected. The left-to-right motion and right-to-left motion are indicated with corresponding arrows.

Motion directionality detection. In the second experiment presented, the goal is to utilize the video representation model for the detection of events involving motion, and the recognition of the directionality of each such event. The scenario used is a sequence with moving cars on a static background, as shown in Figure 4. A 21 frame sequence is used. The objective is to detect the cars as they enter the field-of-view and to distinguish left-to-right from right-to-left motion. The model size used is 37 Gaussians. The detection and identification of the events are based on thresholding as follows: $R_{t,x} \geq 0.3$ detects motion and identifies as positive, the motion from left-to-right. Similarly, $R_{t,x} \geq -0.3$ is the rule used to detect motion in the right-to-left direction. In the presented results (bottom row), the moving objects (dynamic blobs) are detected. The

left-to-right motion and right-to-left motion are indicated with corresponding arrows. The output results accurately reflect the scene dynamics.

Using the space-time blob characteristics the system is able to automatically detect the moving objects in the field of view and to recognize the objects' directionality. The shot is analyzed globally (as a single entity) to extract the representative model. Within the model, each blob's temporal characteristics provide the set of frames within which a moving region is present, from its appearance to its disappearance. The framework proposed is able to cope with short-term occlusion scenarios. As long as there is a sufficient pixel representation for a particular object, before and after the occlusion, the learning phase is able to cluster the feature space into a single cluster. The global analysis thus enables to associate a single blob to the occluded object, therefore overcoming the occlusion.

6 Conclusions and Future Work

In this paper we have described a novel uniform approach for video representation and space-time segmentation of video data. Unsupervised clustering, via Gaussian mixture model (GMM), enables the extraction of video segments, or space-time blobs. An interesting differentiation from existing work in video, is that space and time are treated uniformly, and the video is treated as a single entity as opposed to a sequence of separate frames.

The modeling and the segmentation are combined to enable the extraction of video-regions that represent coherent regions across the video sequence, otherwise termed video-objects or sub-objects. Coherency is achieved in the combined feature space, currently consisting of color, spatial location and time. If motion characteristics on a pixel level are available as a-priori information (e.g. via optical flow), they can be integrated within the proposed framework, as an additional feature (two additional dimensions). Other features, such as texture, shape, etc. can be similarly added to augment region characteristics.

Extracting video regions provides for a compact video content description, that may be useful for later indexing and retrieval applications. Video events are detected and recognized using the GMM and related video-regions. Some experimental results are provided to demonstrate the feasibility of our approach. Each example can be developed into a particular application domain (for example, direction detection for automatic vehicle monitoring). Currently, thresholds were chosen heuristically. Complete systems built with the concepts thus presented will require more in-depth study of the particular application domain, and the relevant initialization procedures, event detection and recognition criteria.

The model proposed assumes a static camera scenario or motion-stabilized frames. The current framework is limited to the description of simple linear motion of a blob, or space-time region, with the analyzed sequence. We therefore

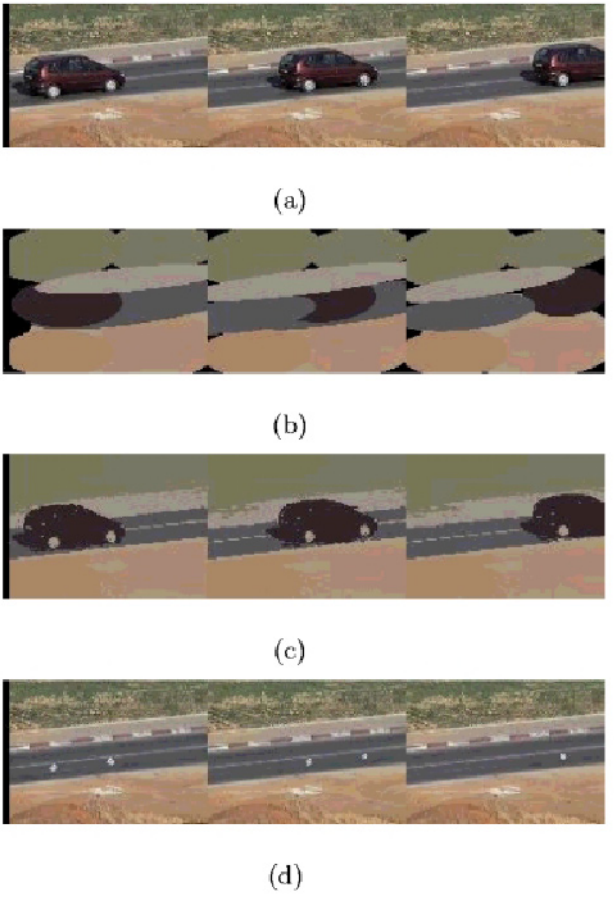

Fig. 5. Blob deletion event. The objective is to detect a moving car and replace the car-body region with static background. (a) The original video data; (b) The sequence representation model; (c) sequence segmentation maps; (d) Output sequence without the moving car-body region. In this experiment an input sequence of 8 frames was used. A 12-Gaussian mixture was used for the representation.

take care to use short time sequences in which motion can be approximated linearly. We are currently extending the methodology to include successive modeling of overlapping short time sequences. This will produce a piece-wise linear approximation of non linear motion trajectories.

Acknowledgment. Part of the work was supported by the Israeli Ministry of Science, Grant number 05530462.

References

1. S. Belongie, C. Carson, H. Greenspan, and J. Malik. Color and texture-based image segmentation using em and its application to content based image retrieval. In *Proc. of the Int. Conference on Computer Vision*, pages 675–82, 1998.
2. R. Castagno, T. Ebrahimi, and M. Kunt. Video segmentation based on multiple features for interactive multimedia applications. *IEEE Trans. on Circuits and Systems for Video Technology*, 8(5):562–571, 1998.
3. S-F Chang, W. Chen, H. Meng, H. Sundaram, and D. Zhong. A fully automated content-based video search engine supporting spatiotemporal queries. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):602–615, 1998.
4. T.M. Cover and J.A. Thomas. *Elements of Information Theory.* John Wiley and Sons,1991.
5. A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Royal Statistical Soc. B*, 39(1):1–38, 1997.
6. Y. Deng and B.S. Manjunath. Content-based search of video using color, texture and motion. In *Proc. IEEE Int. Conf. Image Processing*, volume 2, pages 534–537, 1997.
7. Y. Deng and B.S. Manjunath. Netra-v: Toward an object-based video representation. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):616–627, 1998.
8. B. Duc, P. Schroeter, and J. Bigun. Spatio-temporal robust motion estimation and segmentation.In *6th Int. Conf. Comput. Anal. Images and Patterns*,pages 238–245, 1995.
9. R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* John Wiley and Sons Inc., 1973.
10. A. Hampapur et al. Virage video engine. In *Proc. SPIE*, volume 3022, pages 188–200, 1997.
11. B. Horn and B. Schunck. Determining optical flow. *Artificial Intell.*, 17:185–203, 1981.
12. G. Iyengar and A.B. Lippman. Videobook: An experiment n characterization of video. In *Proc. IEEE Int. Conf. Image Processing*, volume 3, pages 855–858, 1996.
13. A. Jepson and M. Black. Mixture models for optical flow computation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition* , pages 760–761, 1993.
14. V. Koble, D. Doermann, and K. Lin. Archiving, indexing, and retieval of video in the compressed domain. In *Proc. SPIE*, volume 2916,pages 78–89, 1996.
15. C.W. Ngo, T.C. Pong, H.J. Zhang, and R.T. Chin. Motion characterization by temporal slice analysis. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition* , pages 768–773, 2000.
16. P. Salembier and F. Marques. Region-based representations of image and video: Segmentation tools for multimedia services. *IEEE Trans. on Circuits and Systems for Video Technology*, 9(8):1147–1168, 1999.
17. J.Y. Wang and E.H. Adelson. Spatio-temporal segmentation of video data.In *SPIE*, volume 2182, pages 120–131, 1994.
18. G. Wyszecki and W. Stiles. *Color Science: Concepts and Methods, Quantitative Data and Formulae.* Wiley, 1982.
19. L. Zelnik-Manor and M. Irani. Event-based analysis of video. In *Proc. of the IEEE Conf.on Computer Vision and Pattern Recognition*, December 2001.

20. H.J. Zhang, Y. Gong, S.W. Smoliar, and S.Y. Tan. Automatic parsing of news video. In *Proceedings of the International Conference on Multimedia Computing and Systems*, pages 45–54, May 1994.
21. H.J. Zhang and S.W. Smoliar. Developing power tools for video and retrieval. In *SPIE: Storage Retrieval Image and Video Databases*, volume II, 2185, pages 140–149, February 1994.