Event-Based Analysis of Video

Lihi Zelnik-Manor **Michal Irani**
Dept. of Computer Science and Applied Math
The Weizmann Institute of Science
76100 Rehovot, Israel
Email:{lihi,irani}@wisdom.weizmann.ac.il

Abstract

Dynamic events can be regarded as long-term temporal objects, which are characterized by spatio-temporal features at multiple temporal scales. Based on this, we design a simple statistical distance measure between video sequences (possibly of different lengths) based on their behavioral content. This measure is non-parametric and can thus handle a wide range of dynamic events. We use this measure for isolating and clustering events within long continuous video sequences. This is done without prior knowledge of the types of events, their models, or their temporal extent. An outcome of such a clustering process is a temporal segmentation of long video sequences into event-consistent sub-sequences, and their grouping into event-consistent clusters.

Our event representation and associated distance measure can also be used for event-based indexing into long video sequences, even when only one short example-clip is available. However, when multiple example-clips of the same event are available (either as a result of the clustering process, or given manually), these can be used to refine the event representation, the associated distance measure, and accordingly the quality of the detection and clustering process.

1 Introduction

Dynamic events can form a powerful cue for analysis of video information, including event-based video indexing, browsing, clustering, and segmentation. Analysis of events [24, 2, 9, 14, 13, 6, 16, 10] has primarily focused on the recognition of sets of predefined events or actions, or assumed restricted imaging environments. For example, the work of [9] models and recognizes articulated motions, [2] treats facial expressions, and the approaches of [16] and [10] are designed to detect periodic activities. These methods propose elegant approaches for capturing the important characteristics of these events/actions by specialized parametric models with a small number of parameters. These parametric models usually give rise to high-quality recognition of the studied actions. The construction of these parametric models is usually done via an extensive learning phase, where many examples of each studied action are provided (often manually segmented and/or manually aligned).

However, real-world applications are unlikely to be restricted only to recognition of pre-studied carefully modeled events. When dealing with general video data (e.g., movies), often there is no prior knowledge about the types of events in the video sequence, their temporal and spatial extent, or their nature (periodic/non-periodic). A desired application might be for the user who is viewing a movie (e.g., a sports movie), to point out an interesting video segment which contains an event of interest (e.g., a short clip which shows a tennis serve), and request the "system" to fast-forward to the next clip (or find all clips) where a "similar" event occurs. We refer to this as *event-based video indexing* (or "Intelligent Fast-Forward"). Such applications require developing a notion of event-based similarity which is based on a less-specialized (and less restrictive) approach to event modeling.

We regard an event as a stochastic temporal process, where local features at multiple temporal scales are taken as samples of the stochastic process, and are used to construct an empirical distribution associated with this event. The distance between empirical distributions provides a simple statistical distance measure between video sequences (possibly of different lengths) based on their behavioral content. This measure is non-parametric and can thus handle a wide range of dynamic events. This measure may not be optimal for a specific action, but allows for general event-based analysis of video information containing unknown event types.

Having an event-based distance measure between sequences, we can use it for isolating and clustering events within long continuous video sequences. This is done without prior knowledge of the types of events, their models, or their temporal extent. An outcome of such a clustering process is a temporal segmentation of a long video sequence into event-consistent sub-sequences, and their grouping into event-consistent clusters. This is different from the standard temporal segmentation into "scenes" or "shots" (e.g., [12, 25, 18, 11]), which is based on scene-cut or shot-cut detection. Unlike [17], our approach provides temporal segmentation into rich non-atomic actions and events.

While our event-based distance measure is inferior in accuracy to the more specialized (but more restricted) para-

metric models (e.g., [24, 2, 9]), it can be refined with the gradual increase in knowledge about the underlying data. This gives rise to a stratified approach to event-based detection and indexing: When only *one* short example clip of the event-of-interest is available, the simple and crude measure can be used for event-based indexing and detection. However, when multiple example clips of the same event are available (either as a result of the clustering process, or pointed-out manually), these can be used to refine our event representation, the associated distance measure, and the quality of the detection and clustering process.

2 What is an Event?

Events are long-term temporal objects, which usually extend over tens or hundreds of frames. Polana and Nelson [16] separated the class of temporal events into three groups and suggested separate approaches for modeling and recognizing each: (i) *temporal textures* which are of indefinite spatial and temporal extent (e.g., flowing water), see [15] , (ii) *activities* which are temporally periodic but spatially restricted (e.g., a person walking), see [16], and (iii) *motion events* which are isolated events that do not repeat either in space or in time (e.g., smiling). In this paper we refer to *temporal events* as all of the above, and would like to treat all of them within a single framework.

Temporal objects (events) and *spatial objects* bare many similarities as well as differences. Spatial objects are usually characterized by multiple spatial scales [7, 3, 19]. Similarly, temporal objects (events) are characterized by multiple temporal scales. For example, in a sequence of a walking person, the high temporal resolutions will capture the motion of the arms and legs, whereas the low temporal resolutions will mostly capture the gross movement of the entire body.

However, there is a major difference between spatial and temporal objects. Due to the perspective nature of the projection in the spatial dimension, a spatial object may appear at different spatial scales in different images (e.g., depending on whether it is imaged from far or near). In contrast, a temporal event is always characterized by the same temporal scales in all sequences. This is due to the "orthographic" nature of the projection along the temporal dimension (which is simply the temporal sampling at constant frame rate). For example, a single step of a walking person, viewed by two different video cameras of the same frame rate, will extend over the same number of frames in both sequences, regardless of the internal or external camera parameters. Hence, the same event will be captured at the same temporal scales in different sequences, even when viewed from different distances, different viewing positions, or at different zooms. This observation has motivated us to represent and analyze events by performing measurements and comparing them at corresponding temporal scales across different sequences.

3 An Event-Based Distance Measure

Based on the above observations, local features at multiple *temporal scales* of the video sequence are taken as samples of a stochastic temporal process (the event), and are used to construct an empirical distribution associated with this event at each temporal scale. Two events are considered similar if they could have been generated by the same stochastic process, i.e., if their empirical distributions at corresponding temporal scales are similar.

For obtaining features at multiple temporal scales we first construct a *temporal pyramid* of the entire video sequence by blurring and sub-sampling the sequence along the temporal direction only. The temporal pyramid of a sequence S is thus a pyramid of sequences $S^1(=S), S^2, \ldots, S^L$, where the image frames in all the sequences are of the same size, and each sequence S^l has half the number of frames of the higher resolution sequence S^{l-1}. We usually use 3 or 4 temporal scales (i.e., L=3 or 4).

For each sequence S^l in the temporal pyramid, we estimate the local intensity gradient (S^l_x, S^l_y, S^l_t) at each space-time point (x, y, t). The gradient is normal to the local spatio-temporal surface generated by the event in the space-time sequence volume (at temporal resolution l). The gradient direction captures the local surface orientation which depends mostly on the local behavioral properties of the moving object, while its magnitude depends primarily on the local photometric properties of the moving objects and is affected by its spatial appearance (e.g., color and texture of clothes). To preserve the orientation (behavioral) information alone and eliminate as much of the photometric component as possible (the magnitude), we normalize the spatio-temporal gradients to be of length 1. To be invariant to negated contrasts between foreground and background (e.g., a person wearing dark/light clothes against a light/dark background) and to the direction of action (e.g., walking right-to-left or left-to-right), we further take the absolute value of the normalized space-time gradients. Our local space-time feature measurements are therefore:

$$(N^l_x, N^l_y, N^l_t) = \frac{(|S^l_x|, |S^l_y|, |S^l_t|)}{\sqrt{(S^l_x)^2 + (S^l_y)^2 + (S^l_t)^2}} \qquad (1)$$

We associate with each event a set of $3L$ empirical distributions $\{h^l_k\}$ for each feature component $(k = x, y, t)$ at each temporal scale $(l = 1, \ldots, L)$. These empirical distributions capture the statistics of the spatio-temporal shape generated by the event.

Unlike [4], which measures motion features at multiple *spatial* scales, our measurements are performed at multiple *temporal* scales. We therefore capture *temporal textures*[1]

[1] Although [16, 22] used the term "temporal textures", in fact they did not measure texture properties at multiple temporal scales. [20] used the

as opposed to "moving spatial textures" which are captured by [4]. Spatio-temporal textures have been used by [1] for video synthesis. While in video synthesis it is important to preserve both the spatial and the temporal properties of the texture (in order to generate a long realistic looking sequence from a short clip), in event recognition and detection we do not want to be sensitive to the spatial texture, only to the temporal texture properties. Insensitivity to spatial texture is necessary in order to detect different people wearing different clothes as performing the same dynamic operations. Our normalized spatio-temporal features (unlike those of [1] and [4]) are relatively insensitive to the changes in spatial properties of the acting person or of the background.

To illustrate this, Fig.1 shows the empirical distributions of one feature component N_k^l ($k = t, l = 1$) for 6 different clips. Three clips show walking activities performed by different people wearing different clothes, viewed from different viewing angles, from different distances and with different backgrounds. The other three clips show the same person performing different activities (while wearing the same clothes, same background, etc.). The distributions of all the 'walking' clips (marked in blue) are much closer to each other than to those corresponding to the other activities (marked in different colors).

To measure a distance between two sequences we measure the distances between corresponding empirical distributions of all feature components at all temporal scales using χ^2 divergence, and add these to obtain a single (squared) distance measure between the two sequences:

$$D^2 = \frac{1}{3L} \sum_{k,l,i} \frac{\left[h^l_{1_k}(i) - h^l_{2_k}(i)\right]^2}{h^l_{1_k}(i) + h^l_{2_k}(i)} \qquad (2)$$

where the empirical distribution of measurements N_k^l is represented by a discrete smoothed histogram h_k^l whose integral is normalized to 1. Histogram smoothing decreases the sensitivity of the χ^2 test to small local miss-matches between the histograms. The normalization of the integral to 1 gives rise to similar histograms for sequences displaying the same event even when they are of different temporal lengths or of different spatial sizes.

In general, enforcing joint occurrence of features at multiple temporal scales should provide a better distance measure than Eq.(2). However, this requires the use of multi-dimensional histograms (e.g., [19]). These are computationally intensive and memory-consuming (e.g., for $k = 3$, $L = 4$, and assuming 256 bins for each histogram dimension, the size of the mutli-dimensional histogram is 256^{12}). Instead we use single-dimensional histograms and require the joint occurrence of feature *distributions* at multiple temporal scales. The use of single-dimensional histograms reduces the data size to $12 \cdot 256$, which is easy to manage and computationally fast.

Fig. 2 shows the effectiveness of the distance measure of Eq. (2) for event detection and indexing based on a single example clip. A short example clip (showing a basketball player throwing the ball at the basket) was compared against a sliding window continuously shifted across a long video sequence. The long sequence includes actions like dribbling, dunking, etc. Low values indicate temporal regions in the long sequence with high similarity to the example clip. The blue bars on the time axis mark the (ground-truth) video segments in which the player threw the ball at the basket.

Our simplistic (but general) event-based representation and distance measure are probably inferior in accuracy to the more sophisticated (but more restricted) parametric approaches (e.g., [24, 2, 9]). However, they can handle a wide range of unknown events and actions. In Section 5 we discuss how an improved representation and distance measure and can be obtained with the gradual increase of information about the underlying data.

4 Event-Based Clustering

Having an event-based distance measure between sequences, we can use it for isolating and clustering events within long continuous video sequences. This is done without prior knowledge of the types of events, their models, or their temporal extent. An outcome of such a clustering process is a temporal segmentation of a long video sequence into event-consistent sub-sequences, and their grouping into event-consistent clusters.

We use a sliding temporal window to compare every sub-sequence of length T to all other sub-sequences of the same length within a long video sequence. We construct as affinity (similarity) matrix M whose entries are $M(i,j) = M(j,i) = \exp\left[-D_{ij}^2/\sigma\right]$, where D_{ij} is the distance between sub-sequence i and sub-sequence j computed using Eq. (2), and σ is a constant scale factor used for stretching values (see [23] for more details). We then use the normalized-cut approach of [23] (which builds on top of [21]), to cluster the data. The initial clustering is then refined by re-classifying all sub-sequences using cluster representatives (see Section 5).

Figure 3 displays the results of applying the above clustering method to a several-minutes long (approx. 6000 frames) video sequence recorded outdoors by a stationary video camera (see Fig. 3.a - 3.f). We used a temporal window of size $T = 64$ frames, and skips of 8 frames when sliding the window within the long sequence. The sequence contains four types of frequently occurring activities: walking, jogging, hand-waving, and walking-in-place (performed by different people of both genders wearing different clothes for different lengths of time), and single oc-

term "video textures" with a different meaning - for synthesizing video by temporally shuffling frames.

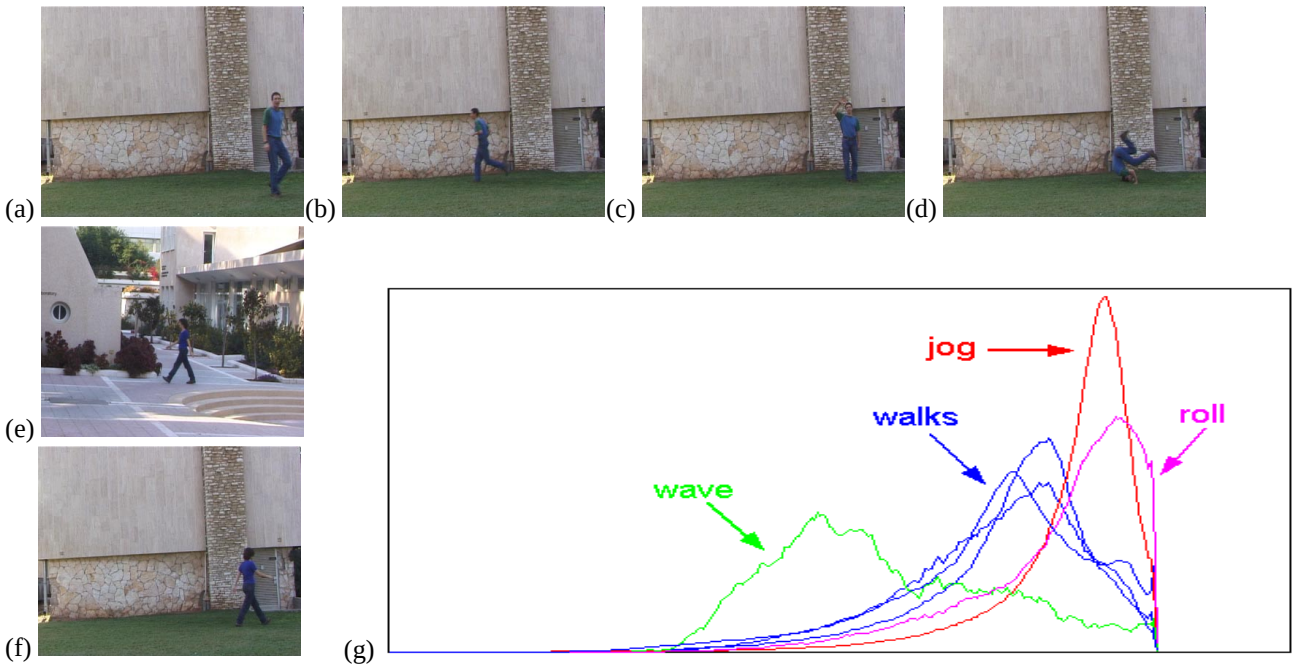

Figure 1: *The empirical distiburion of N_k^t for 6 different clips (see text).*

currences of several other activities (e.g., rolling, and other free activities). Most of the walking is performed parallel to the image plane, but several parts include walking in slightly diagonal directions and some on snake-like paths. Waving includes waving with a single hand or both hands (not necessarily having the same phase). We ignored all space-time points (x, y, t) for which the temporal derivative is below some threshold, thus performing the statistics mostly on spatio-temporal points which participate in the event. This step can be regarded as a very rough spatial segmentation, and was sufficient for our purposes.

Fig.3.g shows the outer product of 5 eigenvectors corresponding to the 5 most dominant eigenvalues of the affinity matrix M, assuming 5 clusters (this is the "Q matrix" of [23]). The rows of M are displayed in their original temporal order. The rows and columns of Q were then sorted in descending order of similarity, resulting in a roughly block diagonal matrix. Representative rows from the first 4 blocks were then used to further refine the clustering (see Section 5). Fig.3.h displays the resulting clustering of the rows of the matrix in Fig.3.g. The rows are no longer temporally ordered as they are now grouped in clusters. The matrix is now block-diagonal, which implies good clustering. The first 4 clusters correspond to the 4 most frequently occurring events in the sequence, while in the fifth cluster fall all the other free events (except for a few misclassifications). To evaluate the quality of the results we visually display the clustering results by color coding the time axis with the respective cluster association (top colored-bar of Fig.3.i). These results can be compared against the ground-truth (manual) classification (bottom colored-bar of Fig.3.i). Almost all the sub-sequences were clustered correctly, and good temporal segmentation into event-consistent sub-sequences (of various lengths) was obtained.

Fig. 4 shows the result of applying event-based clustering to a 500-frame long tennis sequence recorded with a panning camera (here the temporal window was of size $T = 10$ frames). The sequence was first stabilized to compensate for camera motion using [8]. The 3 detected clusters correspond to *strokes* (backhand and forehand, which are clustered together since our local measurements are insensitive to mirror reflections of the same action), *hops*, and *steps* of the tennis player. Fig. 4.f shows clustering result vs. ground-truth.

5 Refining the Representation and Measure

When only a single example clip of each event E is available, the event representation (i.e., the empirical distributions of feature components at multiple temporal scales) is constructed from the single example clip, and the distance between two event clips is estimated using the χ^2 test (see Eq. (2)).

However, when multiple example clips of the same event E are available (either given manually, or obtained via the clustering process), we can refine the event representation and the distance measure to emphasize the contribution of important features at the important temporal scales, as learned from the examples.

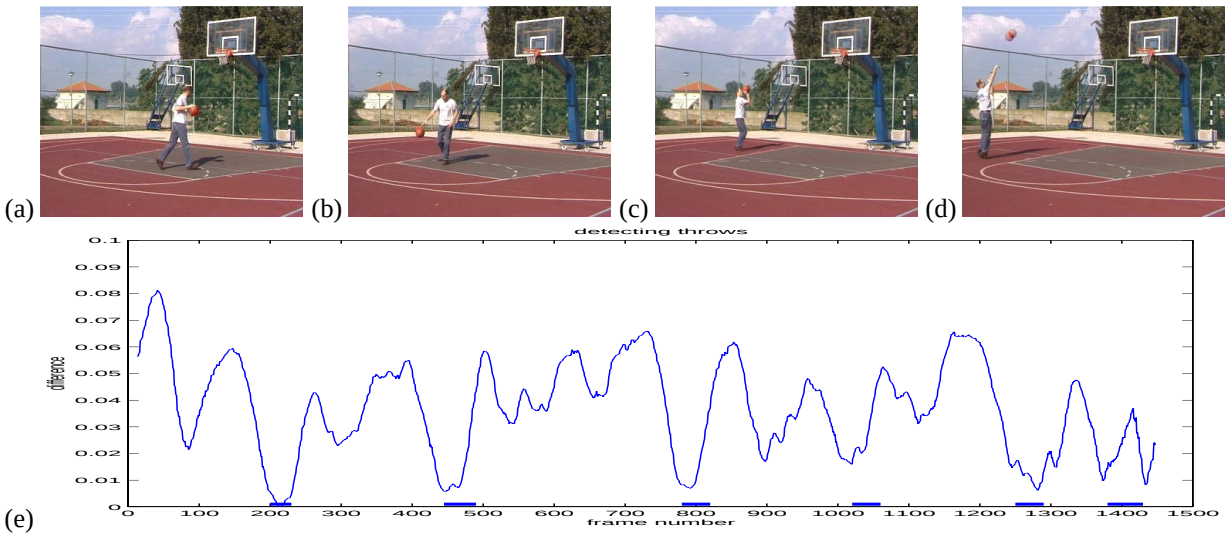

Figure 2: *(a)-(d) Sample frames from a basketball video sequence. (e) Measured distances (using Eq. (2)) between a single throw clip and a sliding window shifted across the entire sequence. See text for more details.*

To show this, we first rewrite the distance measure as a weighted average of the squared differences between two empirical distributions h_1 and h_2:

$$\chi^2 = \sum_i \frac{(h_2(i) - h_1(i))^2}{h_2(i) + h_1(i)} = \sum_i w_i (h_2(i) - h_1(i))^2. \quad (3)$$

The weight w_i assigned to each histogram bin by the χ^2 distance measure is the inverse of the sum of the values in the two histograms at that bin, i.e., $w_i = \frac{1}{h_2(i)+h_1(i)}$. Treating each histogram h as a column vector, we can re-write this in vector notation:

$$\chi^2 = (h_2 - h_1)^T \left[diag(h_2 + h_1)\right]^{-1} (h_2 - h_1) \quad (4)$$

(where $diag(h_2 + h_1)$ is a diagonal matrix whose i-th diagonal entry is $h_2(i) + h_1(i)$).

We next show how to refine the weights w_i and the empirical distributions h with the gradual increase in information about the underlying data.

When multiple example clips of the same event type E are available, we compute the *mean* and *variance* of all the corresponding distributions (separately for each histogram bin of each filter response at each temporal scale). The mean histogram $\overline{h_E}$ can be used as the event representation, and the histogram of variances var_E indicates the reliability and hence the relative significance of the individual histogram bins. This is illustrated in Fig. 5. The solid blue line corresponds to a mean histogram $\overline{h_E}$, whereas the dashed green lines define the envelope corresponding to the mean $+/-$ the standard deviation of all histograms of the example clips. Therefore, when estimating the distance measure between the event E (represented by $\overline{h_E}$) and any new incoming sequence with an empirical distribution h, the weights of Eq. (3) should be replaced with $w_i = \frac{1}{var_E(i)}$ (where $var_E(i)$ is the variance at bin i). Namely, high weights are assigned to bins of low variance (which are more trusted, e.g., near the cyan arrow) and low weights to bins of high variance (which are less reliable, e.g., near the magenta arrow). The refined distance measure specialized for detecting events similar to E is therefore:

$$D_E^2 = \left(h - \overline{h_E}\right)^T \left[diag(var_E)\right]^{-1} \left(h - \overline{h_E}\right) \quad (5)$$

This measure identifies and emphasizes the contribution of prominent spatio-temporal feature components at their prominent temporal scales. Note that for each event type E there will be a different set of weights.

When neighboring histogram bins (which correspond to similar filter responses) are not statistically independent, we can further generalize the distance measure of Eq. (5) by incorporating covariance information and not only the variance:

$$D_E^2 = \left(h - \overline{h_E}\right)^T {cov_E}^{-1} \left(h - \overline{h_E}\right) \quad (6)$$

This is actually the squared mahalanobis distance [5].

Fig. 6 compares the quality of the χ^2-based distance measure of Eq. (2) and the refined distance measure of Eq. (5) for detection purposes. The detection is based on the measured distance between a single (64 frames long) example clip of a walking event compared against a sliding window (of 64 frames) which was shifted across a few-minute-long (6000 frames long) video sequence (the sequence of Fig. 3). The bottom colored-bar marks the ground-truth (manually detected walks). The top colored-bar shows the results using the distance measure of Eq. (2). Even though the example clip contains only a single person walking in a single direction and wearing a particular set of clothes,

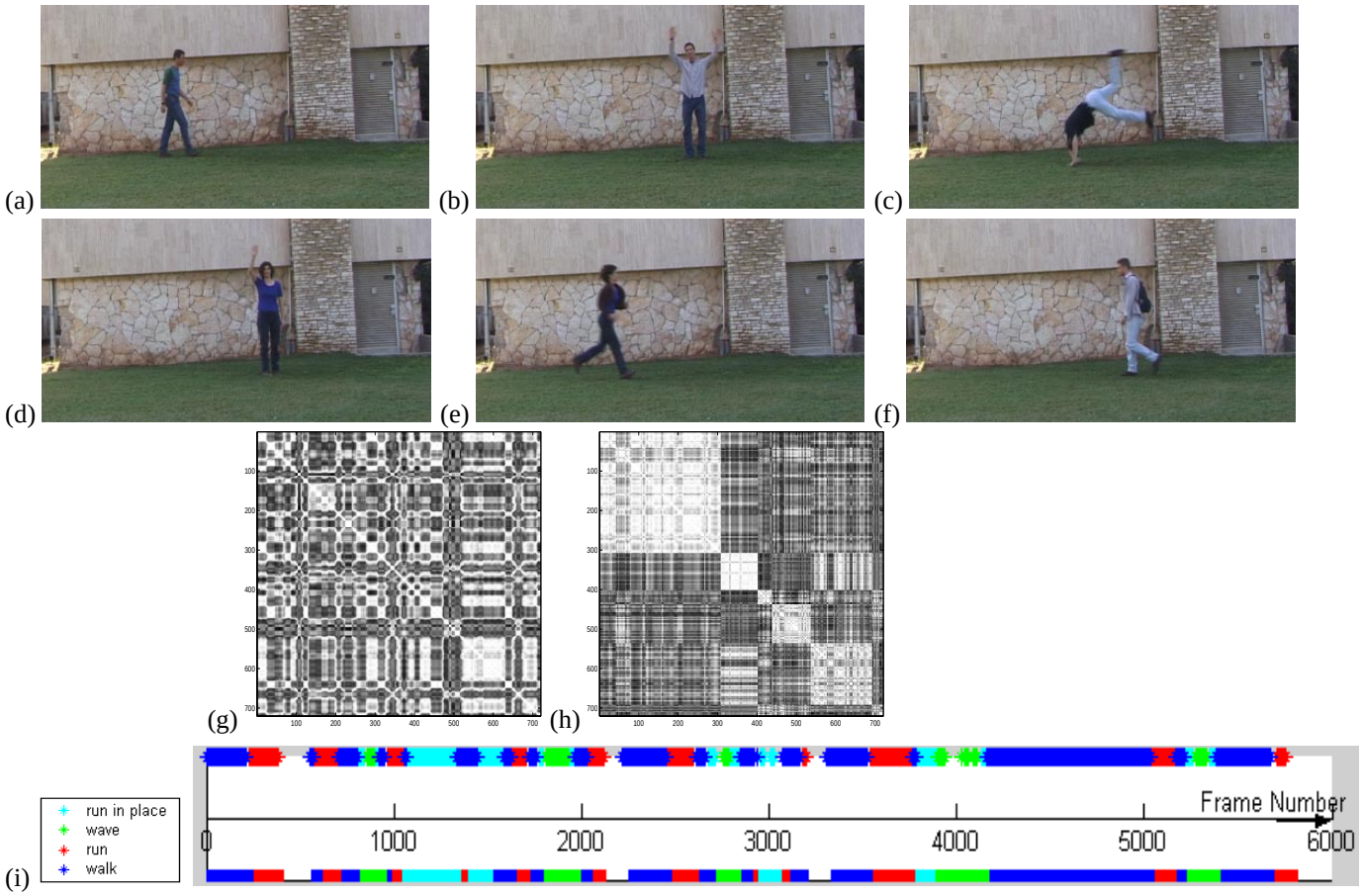

Figure 3: Event-Based Clustering: *(a)-(f) Representative frames of a 6000 frame long video sequence. (g) The "Q" matrix. (h) The clustered matrix. (i) Clustering results displayed on the time axis (top colored bar) vs. ground-truth information (bottom colored bar). See text for more details. For a short segment of the long sequence with clustering results (in B/W to save space),* **see the attached sequence Outdoor.mpg.**

all the other walking people wearing different clothes and walking in different directions were detected. Although there are several false detections, the result indicates that our initial choice of representation and distance measure are reasonable given no other information. The middle colored-bar shows the detection results using the refined distance measure of Eq. (5) based on 10 example clips of walking events (each 64 frames long). Using the refined distance measure significantly reduces the number of false detections.

5.1 A Bayesian Point of View

The problem of event detection can be reposed as follows: Given a new video clip S and an event type E, what is the posteriori probability $P(E|S)$?

According to Bayes rule $P(E|S) = \frac{P(S|E)P(E)}{P(S)}$. When no information is available about the set of possible events, the number of events, or their frequency of occurrence, we assume that all events E are equally likely (i.e., $P(E)$ is the same of all E). Similarly, when no information is available about the types of sequences, we assume that all sequences S are equally likely (i.e., $P(S)$ is the same of all S). In that case $P(E)/P(S)$ is constant, and $P(E|S) \propto P(S|E) = \frac{1}{(2\pi)^{r/2}|Cov_E|^{\frac{1}{2}}} exp\left[-\frac{1}{2}\left(h-\overline{h_E}\right)^T cov_E^{-1}\left(h-\overline{h_E}\right)\right]$, where h denotes the empirical distribution of feature responses in S and r is the dimension of h (i.e., the number of bins). The log-likelihood of E is proportional to the (negated) squared mahalanobis distance defined in Eq. (6). Therefore, small distances directly correspond to high likelihood of the event E, and vice versa. When there is an approximate knowledge about the size of the constant $\frac{P(E)}{P(S)}$, this can be used to determine the choice of threshold to be applied to the distance measure of Eq. (6) for the purpose of event detection and event-based indexing. If we further assume independence between histogram bins, then the off-diagonal entries of the covariance matrix become zero, and the discussion above applies then to the distance

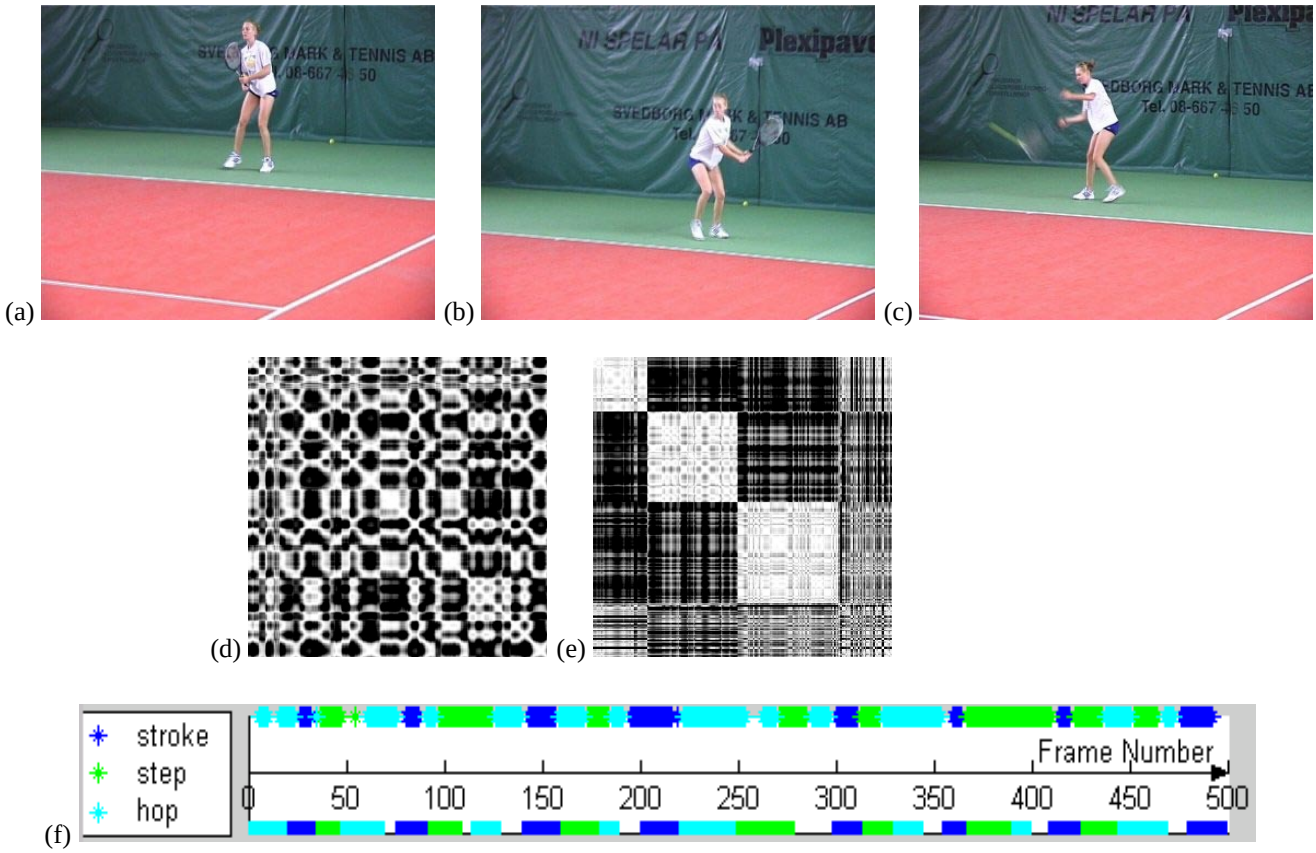

Figure 4: Event-Based Clustering: *(a)-(c) Representative frames of a 500 frame long video sequence. (d) The "Q" matrix. (e) The clustered matrix. (f) Clustering results displayed on the time axis (top colored-bar) vs. ground-truth information (bottom colored-bar). See text for more details. For full sequence with clustering results* **see the attached sequence Tennis.mpg**

measure of Eq. (5). When there is no statistical information about the event E (e.g., when there is only one example clip of event type E) then our probability estimate is based on the standard χ^2 distance measure of Eq.(4).

Let us now examine the case when multiple example clips are available for all event types (e.g., via the clustering process). We can further use our improved distance measures to refine the clustering results, as well as for classification of new incoming sequences. This is done as follows: Let $E_1, ..., E_N$ be the set of all possible events, and let S be a sequence to be classified. Then $P(E_k|S) = \frac{P(S|E_k)P(E_k)}{P(S)} = \frac{P(S|E_k)P(E_k)}{\sum_n P(S|E_n)P(E_n)}$. The priors $P(S|E_1), ..., P(S|E_N)$ can be estimated from the distance measure (the initial or refined, depending on the number or example clips per event type), as explained above. When all events are assumed to be equally likely (i.e., $\forall k \;\; P(E_k) = \frac{1}{N}$), then $P(E_k|S) = \frac{P(S|E_k)}{\sum_n P(S|E_n)}$. Alternatively, one can assume that the prior $P(E_k)$ of an event E_k is proportional to the frequency of its occurrences. This can be estimated by the number of times it was detected in a long video sequence, or by the size of each cluster identified in the clustering process. A sequence S will be classified as event type E_k if $P(E_k|S) = \max_n(P(E_n|S))$. Results of clustering followed by classification refinement according to representative event types are shown in Figs. 3 and 4.

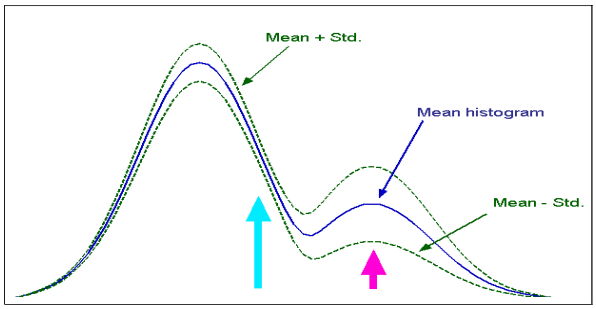

Figure 5: *The mean histogram $\overline{h_E}$ (solid blue line) and the standard deviation from it (green dashed lines). The cyan arrow points at a low variance region and the magenta points at a high variance region.*

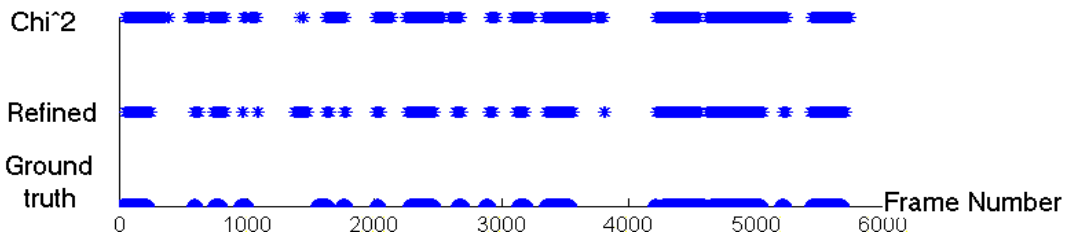

Figure 6: *The results of detecting a walking event within the long (6000 frames) video sequence of Fig. 3. Top colored bar: results using the distance measure of Eq. (2). Middle colored bar: the results using the refined distance measure of Eq.(5). Bottom colored bar: ground-truth (manually detected and marked). See text for more details.*

6 Conclusion

We proposed a simple statistical distance measure between video sequences based on their behavioral content. This measure is non-parametric, and can thus handle a wide range of dynamic events without prior knowledge of the types of events, their models, or their temporal extent. We use this measure for a variety of video applications, including event-based detection, indexing, temporal segmentation, and clustering of long streams of video sequences.

While our event-based distance measure is inferior in accuracy to the more sophisticated (but more restricted) parametric models, it can be refined with the gradual increase in available information about the underlying data.

References

[1] Z. Bar-Joseph, R. El-Yaniv, D. Lischinski, and M. Werman. Texture mixing and texture movie synthesis using statistical learning. *IEEE Transactions on Visualization and Computer Graphics*, 2001. to appear.

[2] M. J. Black and Y. Yacoob. Recogniziong facial expressions in image sequences using local parametrized models of image motion. *International Journal of Computer Vision*, 25(1):23–48, 1997.

[3] J.S. De Bonet. Multiresolution sampling procedure for analysis and synthesis of texture images. In *SIGGRAPH*, pages 361–368, 1997.

[4] O. Chomat and J. L. Crowley. Probabilistic recognition of activity using local appearnce. In *IEEE Conference on Computer Vision and Pattern Recognition*, Fort Collins, USA, June 1999.

[5] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. John Wiley and Sons, Inc., New York, 2001.

[6] D.M. Gavrila and L.S. Davis. 3-d model-based tracking of humans in action: a multi-view approach. In *IEEE Conference on Computer Vision and Pattern Recognition*, SF, 1996.

[7] D.J. Heeger and J.R. Bergen. Pyramid-based texture analysis/synthesis. In *SIGGRAPH*, pages 229–238, 1997.

[8] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision*, 12:5–16, February 1994.

[9] S. X. Ju, M. J. Black, and Y. Yacoob. Cardboard people: A parametrized model of articulated image motion. In *2nd Int. Conf. on Automatic Face and Gesture Recognition*, pages 38–44, Killington,Vermont, Oct. 1996.

[10] F. Liu and R. W. Picard. Finding periodicity in space and time. In *International Conference on Computer Vision*, Bombay, India, January 1998.

[11] A. Nagasaka and Y. Tanaka. Automatic video indexing and full-video search for object appearances. In *Visual Databas Systems II, IFIP*, 1992.

[12] C.W. Ngo, T.C. Pong, H. Zhang, and R.T. Chin. Detection of gradual transitions through temporal slices analysis. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 36–41, June 1999.

[13] S. A. Niyogi and E. H. Adelson. Analyzing and recognizing walking figures in xyt. In *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, June 1994.

[14] S. A. Niyogi and E. H. Adelson. Analyzing gait with spatiotemporal surfaces. In *Workshop on Non-Rigid Motion and Articulated Objects*, Austin, Texas, November 1994.

[15] R. Polana and R. Nelson. Recognition of motion from temporal texture. In *IEEE Conference on Computer Vision and Pattern Recognition*, Champaign, Illinois, June 1992.

[16] R. Polana and R. Nelson. Detecting activities. In *IEEE Conference on Computer Vision and Pattern Recognition*, New-York, June 1993.

[17] Y. Rui and P. Anandan. Segmenting visual actions based on spatio-temporal motion patterns. In *IEEE Conference on Computer Vision and Pattern Recognition*, June 2000.

[18] D.F. Shu S. Swanberg and R. Jain. Knowledge guided parsing in video databases. In *SPIE*, 1993.

[19] B. Schiele and J.L. Crowley. Probabilistic object recognition using multidimensional receptive field histograms. In *International Conference on Pattern Recognition*, Vienna, Austria, August 1996.

[20] A. Schodl, R. Szelsiki, D.H. Salesin, and I. Essa. Video textures. In *SIGGRAPH*, 2000.

[21] J. Shi and J. Malik. Normalized cuts and image segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, San Juan, Puerto Rico, June 1997.

[22] M. Szummer and R. W. Picard. Temporal texture modeling. In *Int. Conf. on Image Processing*, Lausanne, Sep. 1996.

[23] Y. Weiss. Segmentation using eigenvectors: A unifying view. In *International Conference on Computer Vision*, pages 975–982, Corfu, Greece, September 1999.

[24] Y. Yacoob and M. J. Black. Parametrized modeling and recognition of activities. *Computer Vision and Image Understanding*, 73(2):232–247, 1999.

[25] H. Zhang, A. Kankanhali, and W. Smoliar. Automatic partitioning of full-motion video. In *Multimedia Systems*, 1993.