Combination of Accumulated Motion and Color Segmentation for Human Activity Analysis

Alexia Briassouli, Vasileios Mezaris, Ioannis Kompatsiaris
Informatics and Telematics Institute
Centre for Research and Technology Hellas
Thermi-Thessaloniki, 57001, Greece
{abria,bmezaris,ikom}@iti.gr

ABSTRACT

The automated analysis of activity in digital multimedia, and especially video, is gaining more and more importance due to the evolution of higher level video processing systems and the development of relevant applications such as surveillance and sports. This paper presents a novel algorithm for the recognition and classification of human activities, which employs motion and color characteristics in a complementary manner, so as to extract the most information from both sources, and overcome their individual limitations. The proposed method accumulates the flow estimates in a video, and extracts "regions of activity" by processing their higher order statistics. The shape of these activity areas can be used for the classification of the human activities and events taking place in a video and the subsequent extraction of higher-level semantics. Color segmentation of the active and static areas of each video frame is performed to complement this information. The color layers in the activity and background areas are compared using the Earth Mover's Distance, in order to achieve accurate object segmentation. Thus, unlike much existing work on human activity analysis, the proposed approach is based on general color and motion processing methods, and not on specific models of the human body and its kinematics. The combined use of color and motion information increases the method's robustness to illumination

variations and measurement noise. Consequently, the proposed approach can lead to higher-level information about human activities, but its applicability is not limited to specific human actions. We present experiments with various real video sequences, from sports and surveillance domains, to demonstrate the effectiveness of our approach.

I. INTRODUCTION

The analysis of digital multimedia is becoming more and more important as such data is being used in numerous applications, in our daily life, in surveillance systems, video indexing and characterization systems, sports, human-machine interaction, the semantic web and many others. The computer vision community has always been interested in the analysis of human actions from video streams, due to this wide range of applications.

The methods used for the analysis are often application dependent, and they can focus on very particular actions, such as hand gestures [1], [2], sign language, gait analysis [3] [4], or on more general and complex motions, such as exercises, sports, dancing [5], [6], [7], [8]. For specific applications, like gait analysis, kinematic models and models of the human body are often used to analyze the motion, to characterize it (e.g. walking vs. running) and even to identify individuals [9], [10]. In [11] human actions are represented by an appropriate polygon-based model, whose parameters are estimated and fit to a Gaussian mixture model (GMM). Although more general than other methods, this one is dependent on the applicability of the polygon model and the accuracy of the GMM parameter estimation. In other applications, like those concerning the analysis of sports videos [12], the focus is on other cues, namely the particular color and appearance characteristics of a tennis court or a soccer field [13]. Sports-based video analysis also takes advantage of rules in sports, which are very useful for the extraction of semantics from low level features, such as trajectories.

These methods give meaningful results for their respective applications, but have the drawback of being too problem-dependent. The analysis of human actions based on particular models [14], [8] of the human body parts and their motions limits the usability of these methods. For example, a method designed to analyze a video with a side view of a person walking cannot deal with a video of that person taken from a different viewpoint and distance. Similarly, a sports analysis system that uses the appearance of a tennis court or a football field, cannot be used to analyze a different kind of game, or even the same game in a different setting.

Some methods try to avoid these problems by taking advantage of general, spatio-temporal information from the video. Image points with significant variations in both space and time ("space-time interest points") are detected in [15], and descriptors are constructed for them to characterize their evolution over time and space. In [16], "salient points" are extracted over time and space, and the resulting features are classified using two different classifiers. These systems are not application dependent, but are susceptible to inaccuracies in feature point detection and tracking, and may not perform well with real videos, in the presence of noise. Spatiotemporal point descriptors also have the drawback of not being invariant to changes in the direction of motion [15], so their general applicability is limited. Another common approach to human motion analysis is modeling the human body by blobs [17], [18], and then tracking them. However, these methods are based on appropriately modeling the blobs based on the skin color, and would fail in situations where the skin color is not consistent or visible throughout the sequence. Essentially, they are designed to work only in controlled indoors environments. Finally, other appearance based methods, like [19] are successful in isolating color regions in realistic environments, but suffer from lack of spatial localization of these areas.

In order to design an effective and reliable system for human motion analysis, hybrid approaches need to be developed, that take advantage of the information provided by features like color and motion, but at the same time overcome the limitations of using each one separately. We propose a robust system for the analysis of video, which combines motion characteristics, and the moving entities' appearance. As opposed to [20], we do not resort to background removal, and also avoid the use of a specific human model, which makes our method more generally applicable to situations where the person's appearance or size may change. We do not use a model for the human body or actions, and avoid using feature points, so the proposed method is generally applicable and robust to videos of poor quality. The resulting information can be used for the semantic interpretation of the sequence, the classification and identification of the human activities taking place, and also of the moving entities (people).

The processing system developed in this paper can be divided into three main stages. Initially, we estimate optical flow, and accumulate the velocity estimates over subsequences of frames. In the case of a moving camera, its motion can be compensated for in a pre-processing, global motion estimation stage [21], and our method is applied to the resulting video. An underlying assumption is that the video has been previously segmented into shots, which contain an activity

or event of interest. Since there are not completely new frames in a single shot (e.g. in a sports video, one shot will show the game, but frames showing only the spectators will belong to a different shot), it is realistic to assume that the camera motion can be compensated for. A novel method is then developed to determine which pixels undergo motion during a subsequence, by calculating the statistics of all flow estimates. This results in binary activity masks, which contain characteristic signatures of the activities taking place, and can be immediately incorporated in a video recognition or classification system. This is similar to the idea of Motion Energy Images (MEIs), presented in [7]. However, in that work, MEIs are formed from the union of thresholded inter-frame differences. This procedure is very simple and is not expected to be robust in the presence of measurement noise, varying illumination, camera jitteriness. The approach presented in this paper is compared against results obtained with MEIs to demonstrate the advantages of more sophisticated processing. After the motion processing stage, the shapes of the resulting activity areas (equivalently, MEIs) are represented using shape descriptors, which are then included in an automated classification and recognition application. It should be noted that in [7], Motion History Images (MHIs) are also used for recognition purposes, as they contain information about how recent each part of the accumulated activity is. The incorporation of time-related information regarding the evolution of activities is a topic for future extensions of our proposed method, but has not been included in the present work, as it is beyond its current scope.

The second part of our system performs mean-shift color segmentation of the previously extracted activity and background areas. The color of the background can be used to identify the scene, and consequently the context of the action taking place. At the third stage, we compare the color layers of the background and activity areas using the Earth Mover's Distance. This allows us to determine which pixels of the activity areas match with the background pixels, and thus do not belong to the moving entity. As our experiments show, this comparison leads to accurate segmentation results, which provide the most complete description of the video, since they give all the appearance information available for the moving objects. Finally, all intermediate steps of the proposed method are implemented using computationally efficient algorithms, making our approach useful in practical applications.

This paper is organized as follows. In Section II we describe the motion processing stage used to find the areas of activity in the video. The analysis of the shape of these areas for

the understanding of human activities is described in Section III. Section IV presents the color analysis method used for the color segmentation of each frame. The histogram comparison method used to combine the motion and color results is presented in Section V. Experiments with real video sequences, also showing the intermediate results of the various stages of our algorithm, as well as the corresponding semantics, are presented in Section VI. Finally, conclusions and plans for future work are described in Section VII.

II. Motion Analysis: Activity Area extraction from optical flow

Motion estimation is performed in the spatial domain using a pyramidal implementation of the Lucas Kanade optical flow algorithm, which computes the illumination variations between pairs of frames [22]. Assuming constancy of illumination throughout the video sequence, changes in luminance are expected to originate only from motion in the corresponding pixels [23], [24]. Indeed, the motion estimation stage results in motion vectors in textured regions, and near the borders of the moving objects. However, this alone does not give sufficient information to characterize the motion being performed, or to extract the moving objects [25]. For this reason, we have developed a method based on the accumulation of motion estimates throughout the entire sequence, so as to more fully describe the actions or events taking place.

In reality, the constant illumination assumption of the optical flow methods is not satisfied, since there are always slight illumination changes in a scene, as well as camera instability and measurement noise [26]. As a consequence, these variations in luminance are often mistaken for motion, and the resulting optical flow estimates are noisy. Our approach actually takes advantage of this drawback of optical flow methods, namely of the fact that the velocity estimates between pairs of frames are noisy. We accumulate velocity estimates over a large number of frames, that may be affected by noise from imperfect measurements and illumination variations. There is no prior knowledge about the statistical distribution of measurement noise, however the standard assumption in the literature is that it is independent from pixel to pixel, and follows a Gaussian distribution [27]. In practice, even if the noise is not Gaussian, this approximation is sufficient for our purposes, as explained below, in Eq. (2). Thus, we have the following hypotheses:

$$\begin{aligned} H_0 : v_k^0(\bar{r}) &= z_k(\bar{r}) \\ H_1 : v_k^1(\bar{r}) &= u_k(\bar{r}) + z_k(\bar{r}), \end{aligned} \tag{1}$$

where $v_k^i(\bar{r})$ ($i = \{0, 1\}$) are the flow estimates at pixel $\bar{r}$. Hypothesis H_0 expresses a velocity estimate at pixel $\bar{r}$, in frame k, which is introduced by measurement noise, and hypothesis H_1 corresponds to the case where there is motion at pixel $\bar{r}$, expressed by the velocity $u_k(\bar{r})$, which is corrupted by additive noise $z_k(\bar{r})$ [28].

Since the noise $z_k(\bar{r})$ is assumed to follow a Gaussian distribution, we can detect which velocity estimates correspond to a pixel that is actually moving by simply examining the non-gaussianity of the data [29]. The classical measure of a random variable's non-gaussianity is its kurtosis, which is defined by:

$$\mathbf{kurt}(\mathbf{y}) = \mathbf{E}[\mathbf{y^4}] - \mathbf{3}(\mathbf{E}[\mathbf{y^2}])^{\mathbf{2}}. \tag{2}$$

However, the fourth moment of a Gaussian random variable is $\mathbf{E}[\mathbf{y^4}] = \mathbf{3}(\mathbf{E}[\mathbf{y^2}])^{\mathbf{2}}$, so its kurtosis is equal to zero. It should be emphasized that the kurtosis is a measure of a random variable's Gaussianity, regardless of its mean. Thus, the kurtosis of a random variable with any mean, zero or non-zero, will be zero for Gaussian data, and non-zero otherwise. Consequently, this test allows us to detect any kind of motion, as long as it deviates from the distribution of the noise in the motion estimates. Although the Gaussian model is only an approximation of the unknown noise in the motion estimates, the kurtosis remains appropriate for separating true velocity measurements, which appear as outliers, from the noise-induced flow estimates. In [30], it is proven that the kurtosis is a robust, locally optimum test statistic, for the detection of outliers (in our case true velocities), even in the presence of non-Gaussian noise. This is verified by our experimental results, where the kurtosis obtains significantly higher values at pixels that have undergone motion. In the sequel we give a detailed explanation of how the pixels whose kurtosis is considered equal to zero are chosen.

In order to justify the modeling of the flow estimates for the moving pixels as non-Gaussian and as Gaussian for the static pixels, we conduct experiments on real sequences. We manually determine the area of active pixels in the surveillance sequence of the fight, used in Sec. VI-F, to obtain the ground truth for the activity area. We then estimate the optical flow for all pixels and frames in the video sequence. Using the (manually obtained) ground truth for the activity area, we separate the flow estimates for "active pixels" from the flow estimates of the "static pixels". For this video sequence, consisting of 288×384 frames (total of 110592 pixels per frame), there are 9635 active pixels and 100957 static pixels, in each of the 178 frames examined. We extract

the kurtosis of each pixel's flow estimates based on Eq. (2), where the expectations $E[\cdot]$ are approximated by the corresponding arithmetic means, over the video frames. Fig. 1 shows two plots, one of the kurtosis of the active pixels' flow values, and one of the kurtosis of the static pixels' flow estimates. It is evident from Fig. 1 that the kurtosis of the active pixels obtains much higher values than that of the static pixels. In particular, its mean value over the entire sequence is 1.0498 for the active pixels, 0.0015 for the static ones, while the mean kurtosis for all pixels is equal to 1.0503 (again, this mean is estimated over all pixels, over all video frames). Thus, for this real video sequence, the static pixels' mean kurtosis is equal to 0.001428% of the mean kurtosis of all frame pixels, and 0.001429% of the mean kurtosis of the active pixels.

There is no generally applicable, theoretically rigorous way to determine which percentage of the kurtosis estimates should be considered zero (i.e. corresponding to flow estimates that originate from static pixels), since there is no general statistical model for the flow estimates in all possible videos, due to the vast number of possible motions that exist. Consequently, we *empirically* determine which pixels are static, by examining the videos used in the experiments, and also ten other similar videos (both outdoors sports and indoors surveillance sequences). Similarly to the analysis for Fig. 1, we first manually extract the activity area as ground truth. We then calculate the optical flow for the entire video, and find the kurtosis of the flow estimates for each frame pixel based on Eq. (2), by averaging over all video frames. The mean kurtosis of the flow estimates in the active and static pixels is calculated, and it is found that the mean kurtosis in the static pixels is less than 5% of the mean kurtosis of the active pixels (and 0.047% of the mean kurtosis of all pixels). This leads us to consider that pixels whose average kurtosis of the flow estimates, accumulated over the video sequence, is less than 0.1 of the average kurtosis over the entire video frames, can be safely considered to correspond to static pixels; small variations of this threshold were experimentalaly shown to have little effect on the accuracy of the results.

A similar concept, namely that of Motion Energy Images (MEIs) is presented in [7], where the pixels of activity are localized in a video sequence. This is achieved by thresholding inter-frame differences and taking the union of the resulting binary masks. The activity areas of our approach are expected to lead to better results, for the following reasons:

- Our method processes the optical flow estimates, which are obviously a more accurate and robust measure of illumination variations (motion) than simple frame differencing. It should

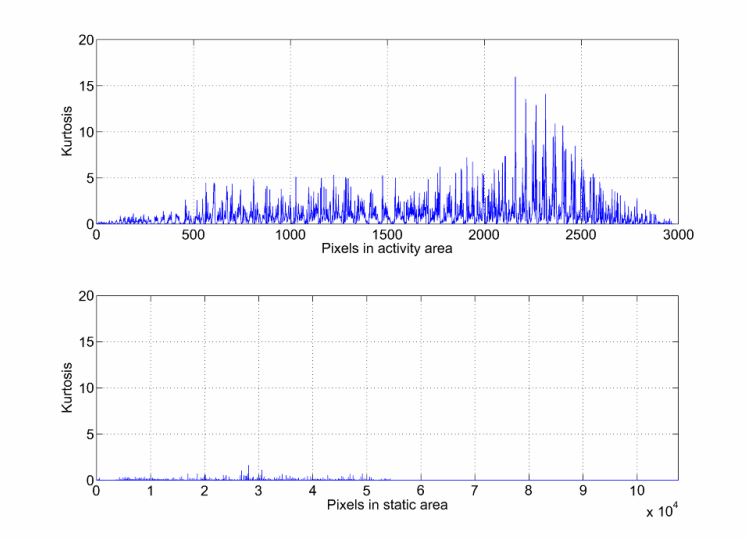

Fig. 1. Kurtosis estimates for the active and static pixels. The activity area and static pixels have been obtained via manual localization, to obtain the ground truth.

be noted that their computation does not incur a significant computational cost, due to the efficient implementations of these methods that are now available.

- Our method processes the optical flow estimates using higher the statistics of the kurtosis, which is is a robust detector of outliers from the noise distribution, as explained above. Since there is no theoretically sound and generally applicable method for determining the threshold for the frame differences (even in [7]) used for MEIs, we determined their optimal thresholds via experimentation. Nevertheless, inaccuracies introduced by camera jitteriness, panning, or small background motions, cannot be overcome in a reliable manner, even when the best possible threshold is chosen empirically, when using simple thresholding of frame differences.

In the experiments of Sec. VI we compare the MEIs of [7] with the activity areas produced by our method, both qualitatively and quantitatively. Indeed, the proposed approach leads to activity areas that contain a more precise "signature" of the activity taking place, and are more robust to measurement noise, camera jitteriness, illumination variations and small motions in the background (e.g. moving leaves). It is also more sensitive to small but consistently appearing

motions, like the trajectory of a ball, which are not found as easily or accurately by the MEI method.

A. Subsequence selection, event detection

An important issue that needs to be addressed is the number of frames that are chosen to be used for the formation of the activity mask. Initially, a fixed number of frames (k) is selected[1], and their accumulated pixel velocities $v_k(\bar{r})$ are denoted by the vector $V_k(\bar{r}) = [v_1(\bar{r}), v_2(\bar{r}), ..., v_k(\bar{r})]$. The flow over new frames is continuously accumulated, and each new value (at frame $k+1$) is compared with the standard deviation of the k previous flow values as follows:

$$v_{k+1}(\bar{r}) \begin{cases} \leq std(V_k(\bar{r})), & \text{continue accumulating frames.} \\ > std(V_k(\bar{r})), & \text{can stop accumulating frames.} \end{cases} \tag{3}$$

Thus, when a new flow estimate is greater than one standard deviation of the k previous estimates, we consider that motion begins at that frame.

To better illustrate the procedure of (3), we analytically present two relevant examples in Fig. 2, where the flow values for a background and a moving pixel from the sequence of Sec. VI-C are compared (this sequence was used in the example of Fig. 1 as well). For Fig. 2(a), the standard deviation of the flow estimates from frames 1 to 21 was equal to 0.342, and the velocity estimate at frame 22 is 5.725, so we conclude that the pixel starts moving at frame 22 (this agrees with our ground truth observation). On the other hand, the standard deviation of the static pixel is, on average, equal to 0.35, and its velocity never becomes higher than 0.5. Similarly, in Fig. 2(b) the standard deviation of the flow estimates until frame 31 is 0.479, and the flow estimate at frame 32 "jumps" to 16.902, making it evident that the (active) pixel starts moving at frame 32.

In Fig. 2(a), there are some fluctuations of the flow between frames $23 - 32$, which may introduce a series of "false alarm" beginnings and endings of events (e.g. at frame 29 the flow estimate is 0.2, i.e. lower than the standard deviation of the previous frames, which is equal to 4.24, indicating an end of activity). However, these are eliminated via post-processing, by setting a threshold of k for the duration of an event, i.e. we consider that no motion can begin/end during 10 frame subsequences. This sets a "minimum event size" of 10 frames, which does not create

[1]In the experiments, $k = 10$ is chosen empirically. The sequences examined here have at least 60 frames, and in practice videos are much longer, so this choice of k is realistic.

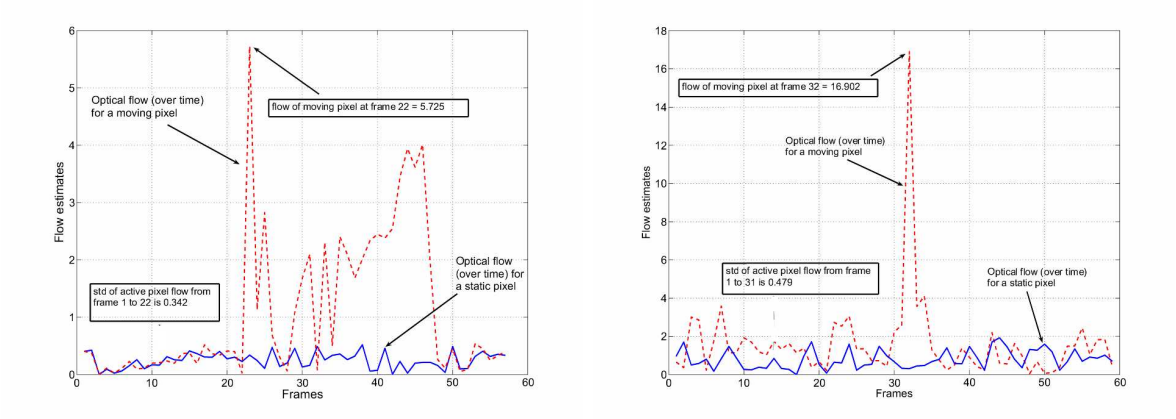

Fig. 2. Optical flow values for a moving and a background pixel over time (video frames). The value of the optical flow of the moving pixel at the frame of change is significantly higher than the standard deviation of its flow values in the previous frames, whereas the value of the optical flow of a static pixel at all frames remains comparable to its flow values in the previous frames.

problems in the activity area extraction, since, in the worst case, frames with no activity will be included in an "active subsequence", which cannot degrade the shape of the actual activity region. In this example, we consider that there is no new event (beginning or ending) until frame 32. After frame 33, the values of the flow are comparable to the standard deviation of the previous flow estimates, so we consider that the pixel remains active. However, at frame 47, the pixel flow drops to 0.23, while the previous flow value was 4. In order to determine if the motion has stopped, we then examine the flow values over the next 10 frames. Indeed, from frames 47 to 57 the standard deviation of the flow estimates is 0.51, and the flow values are comparable. Thus, we can consider that the subsequence of that particular pixel's activity ends at frame 47.

Similar experiments were conducted with the videos used in Sec. VI, and ten similar indoors and outdoors sequences, where the start and end times of events were determined according to Eq. (3) and this procedure. The results were compared with ground truth, extracted by observing the video sequences to extract the begin and end times of events, and led to the conclusion that this is a reliable method for finding when motions begin and end.

Once a subsequence containing an event has been selected, we accumulate the noisy inter-

frame velocity estimates of each pixel over those frames, and estimate their kurtosis, as described in the previous section. The pixels whose kurtosis is higher than 0.1 times the average subsequence kurtosis are considered to belong to an object that has moved over the frames that we are examining. Examples of the resulting activity areas are shown in Fig. 3 (c)-(e), where it is obvious that the moving pixels are correctly localized, and, more importantly, that the resulting areas have a shape that is indicative of the event taking place. These activity areas can be particularly useful for the extraction of semantic information concerning the sequence being examined, when, for example, they are characterized by a shape representative of specific actions. This also is evident in our experiments (Sec. VI), where numerous characteristic motion segments have been extracted via this method.

III. Human Action Analysis from Activity Areas

The activity areas extracted from the optical flow estimates (Sec. II) contain the signatures of the motions taking place in the subsequence being examined. The number of non-zero areas gives an indication of the number of moving entities in the scene. In practice, the number of non-zero areas is greater than the number of moving objects, due to the effects of noise. However, this can be dealt with by extracting the connected components, i.e. the actual moving objects in the activity areas, via morphological post-processing.

For example, in Fig. 3(c)-(e) we show the activity areas extracted for various phases of a tennis hit, which has been filmed from a close distance (Fig. 3 (a), (b)). The different parts of the arm and leg movement create distinct signatures in the resulting activity masks for each subsequence. After accumulating the first ten frames (Fig. 3 (a)), we can discern the trajectory of the ball, which is "approaching" the tennis player. Fig. 3(b) shows the activity area when the tennis ball has actually reached the player. This information, combined with prior knowledge that this is a tennis video, can lead us to the conclusion that this is a player receiving the ball from the tennis serve. This conclusion can be further verified by the activity area resulting from the processing of frames 1 to 30, shown in Fig 3(c). In this case, one can see the entire ball trajectory, before and after it is hit, from where one can conclude that the player successfully hit the ball. Naturally, such conclusions cannot be arbitrarily drawn for any kind of video with no constraints whatsoever. As is the usual case in systems for recognition, sports analysis [13], modeling of videos [31], some prior knowledge is necessary to extract semantically meaningful

conclusions, at a higher level. In this case, knowledge that this is a sports video can lead to the conclusion that the trajectory most probably corresponds to a ball. Additional knowledge that this is a tennis video allows us to infer that the ball reaches and leaves the player, and that consequently the player successfully hit the ball.

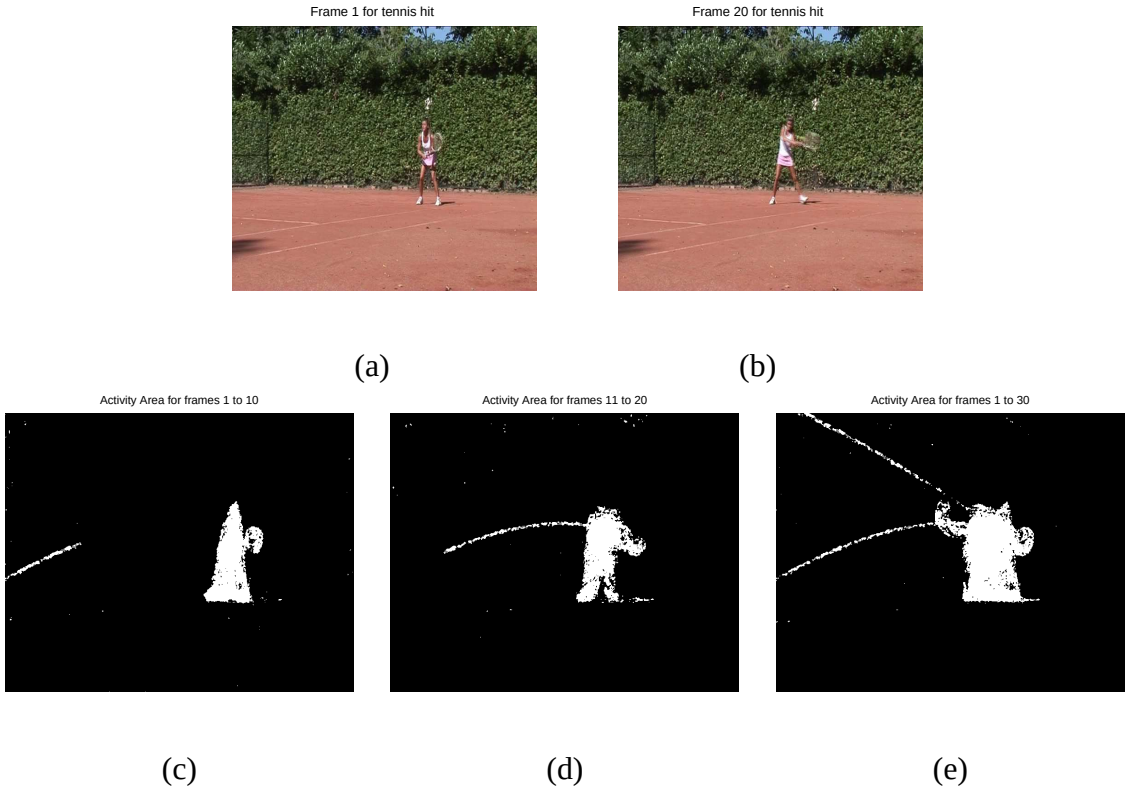

Fig. 3. Tennis hit: (a) frame 1, (b) frame 20. Activity Areas for Tennis Hit: (a) frames 1-10,. (b) frames 11-20, (c) frames 1-30.

A. Activity Area Shape Extraction and Comparison

Features extracted from a video sequence can be used to characterize the way the players hit the ball, to identify them. In our case, we choose to use shape descriptors, since they contain important characteristics about the type of activity taking place, as seen in Sec. III. For an actual video application, the activity areas can be automatically characterized and subsequently compared with the shape descriptors that are used by the MPEG-7 Standard [32], [33]. We focus on the 2D contour-based shape descriptor [34] to represent the activity areas, since the most revealing information about the events taking place is contained in the contours. This

descriptor is based on the curvature scale-space (CSS) representation [35], and is particularly well suited for our application, as it distinguishes between shapes that cover a similar area, but have different contours. It should be noted that the CSS descriptor used in MPEG-7 has been selected after very comprehensive testing and comparison with other shape descriptors, such as those based on the Fourier transform, Zernike moments, turning angles and wavelets [34].

To obtain the CSS descriptor, the contour is initially sampled at equal intervals, and the 2D coordinates of the sampled points are recorded. The contour is then smoothed with Gaussian filters of increasing standard deviation. At each filtering stage, fewer inflection points of the contour remain, and the contour gradually becomes convex. Obviously, small curvature changes are smoothed out after a few filtering stages, whereas stronger inflection points need more smoothing to be eliminated. The CSS image is a representation which facilitates the determination of the filtering stage at which a contour becomes convex, and its shape becomes smooth. The horizontal coordinates of the CSS image correspond to the indices of the initially sampled contour points that have been selected to represent it, and the vertical coordinates correspond to the amount of filtering applied, defined as the number of passes of the filter. At each smoothing stage, the zero-crossing points of the curvature (where the curvature changes from convex to concave or vice versa) are found, and the smoothing stage at which they achieve their maxima (which appear as peaks in the CSS image), is estimated. Thus, the peaks of the CSS image are an indicator of a contour's smoothness (lower peaks mean that few filtering stages were needed, i.e. that the original contour was smooth). Intuitively, the CSS descriptor calculates how fast a contour turns: by finding the curvature zero-crossing points, we find at which smoothing stage the contour has become smooth. Thus, an originally jagged contour will need more smoothing stages for its curvature zero-crossings to be maximized, than a contour that is originally smooth.

The shape comparison based on CSS shape descriptors follows the approach of [36], [37]. The CSS representation of the contours to be compared consists of the maxima (peaks) of the corresponding CSS images, equivalently the smoothing stage at which the maximum curvature is achieved. In order to compare two contours, possible changes in their orientation first need to be accounted for. This is achieved by applying a circular shift to one of the two CSS image maxima, so that both descriptors have the same starting point. The Euclidean distances between the maxima of the resulting descriptors are then estimated and summed, giving a measure of how much the two contours match. When the descriptors contain a different number of maxima, the

TABLE I

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF TENNIS HIT.

Frames	Smoothed Curvature	Original Curvature	Smoothing Stage for Maximum Curvature
$1-10$	(4,10)	(3,10)	11
$11-20$	(3,13)	(3,12)	21
$1-30$	(3,10)	(3,9)	38

coordinates of the unmatched maxima are also added to this sum of Euclidean distances. This procedure is used in the experiments of Sec. VI-G in order to determine what kind of activity takes place in each subsequence, to measure the recognition performance of the proposed, activity area-based approach, and to compare its performance to that of the Motion Energy Image based method of [7].

In Table I we show the shape descriptors extracted for the activity areas of the video of a tennis hit, shown in Fig. 3. The table shows the curvature of the original and smoothed contours, and the maximum smoothing stage at which there are curvature zero-crossings. In columns two and three, the pairs of numbers correspond to the curvature of the accumulated horizontal (x) coordinates, and vertical coordinates (y) [34]. The curvature has very similar values, both before and after smoothing. This is expected, since the overall shape of the activity area did not change much: the player translated to the left, and also hit the ball. However, the area for frames $1-30$ has a higher zero-crossing peak, which should be expected, since in Fig. 3(c) there is a new curve on the left, caused by the player hitting the ball, and also its new trajectory.

IV. COLOR SEGMENTATION: MEAN SHIFT

In order to fully extract a moving object and also acquire a better understanding of its actions, for example, how a human is walking or playing a sport, we analyze the color information available in it and combine it with the accumulated motion information. The color alone may provide important information about the scene [38], [39], the moving entities, as well as the semantics of the video, e.g. from the color of a tennis court we know if it is grass (green) or clay (red). This paper does not focus on the use of color by itself for recognition or classification purposes, as its aim is to recognize human activities, and thus use color to complement motion

information. When color is combined with the motion characteristics extracted from a scene, we can segment the moving objects, and thus extract additional information concerning the people participating, the kind of activity they are performing, and their individual motion and appearance characteristics. In the proposed method, the usage of color is not sensitive to inter-frame illumination variations, or to different color distributions caused by using different cameras, as the color distribution is compared between different regions of the same frame (see Sec. V).

Color segmentation is performed using the mean shift [40], as it is a general-purpose unsupervised learning algorithm, which makes autonomous color clustering a natural application for it. Unlike other clustering methods [41], mean shift does not require prior knowledge of the number of clusters to be extracted. It requires, however, determining the size of the window, where the search for cluster centers takes place, so the number of clusters is determined in an indirect manner. This also allows it to create arbitrarily shaped clusters, or object boundaries, so its applicability is more general than that of other methods, such as K-means [41]. The central idea of the mean shift algorithm is to find the modes of a data distribution, i.e. to find the distribution's maxima, by iteratively shifting a window of fixed size to the mean of the points it contains [42]. In our application, the data is modelled by an appropriate density function, and we search for its maxima (modes) by following the direction where its gradient increases [43], [44]. This is achieved by iteratively estimating the data's mean shift vector (see Eq. (6) below), and translating the data window by it until convergence. It should be noted that convergence is guaranteed, as proven in [40].

For color segmentation, we convert the pixel color values to L*u*v* space, as distances in this space correspond better to the way humans perceive distances between colors. Thus, each pixel is mapped to a feature point, consisting of its L*u*v* color components, denoted by $\mathbf{x}$. Our data consists of n data points $\{\mathbf{x}_i\}_{i=1,\ldots,n}$, in d-dimensional Euclidean space $\mathcal{R}^d$, whose multivariate density is estimated with a kernel $K(\mathbf{x})$, and window of radius h, as follows:

$$\hat{f}(\mathbf{x}) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right). \tag{4}$$

Here, $d = 3$, corresponding to the dimensions of the three color components. The kernel is chosen to be symmetric and differentiable, in order to enable the estimation of the pdf's gradient, and

consequently its modes as well. The Epanechnikov kernel used here is given by

$$K_E(\mathbf{x}) = \begin{cases} \frac{1}{2}c_d^{-1}(d+2)(1-\mathbf{x}^T\mathbf{x}), \ if \mathbf{x}^T\mathbf{x} < 1 \\ 0, \ \text{otherwise.} \end{cases} \tag{5}$$

It is shown in [40] that, for the Epanechnikov kernel, the window center needs to be translated by the "sample mean shift" $M_h(\mathbf{x})$ at every iteration, in order to converge to the distribution's modes. This automatically leads to the cluster peaks, and consequently determines the number of distinct peaks. The sample mean shift is given by

$$M_h(\mathbf{x}) = \frac{1}{n_x} \sum_{\mathbf{x}_i \in S_h} \mathbf{x}_i - \mathbf{x}, \tag{6}$$

where n_x is the number of points contained in each search area $S_h(\mathbf{x})$. The mean shift is estimated so that it always points in the direction of gradient increase, so it leads to the pdf maxima (modes) of our data. We obtain the color segmentation of each video frame by the following procedure:

- The image is converted into L*u*v* space, where we randomly choose n image feature points $\mathbf{x}_i$. These are essentially n pixel color values.
- For each point $i = 1, ..., n$, we estimate the sample mean shift $M_h(\mathbf{x}_i)$ in a window $S_h(\mathbf{x}_i)$ of radius h around point $\mathbf{x}_i$.
- The window $S_h(\mathbf{x}_i)$ is translated by $M_h(\mathbf{x}_i)$ and a new sample mean shift is estimated, until convergence, i.e. until the shift vector is approximately zero.
- The pixels with color values closest to the density maxima derived by the mean shift iterations are assigned to those cluster centers.

The number of the extracted color clusters is thus automatically generated, since it is equal to the number of the resulting distribution peaks. In Fig. 4 we show a characteristic example of the segmentation achieved by using the Mean Shift Algorithm. The pixels with similar color have indeed been grouped together, and the algorithm has successfully discriminated even between colors which could cause confusion, like the color of the player's skin and the tennis court.

V. Combination of Activity Areas and Color for Moving Object Segmentation

The mean shift process described in the previous section leads to the separation of each frame into color-homogeneous "layers" or regions. The activity areas give the possible locations of the moving entities in each frame, but not their precise location. However, they indicate which

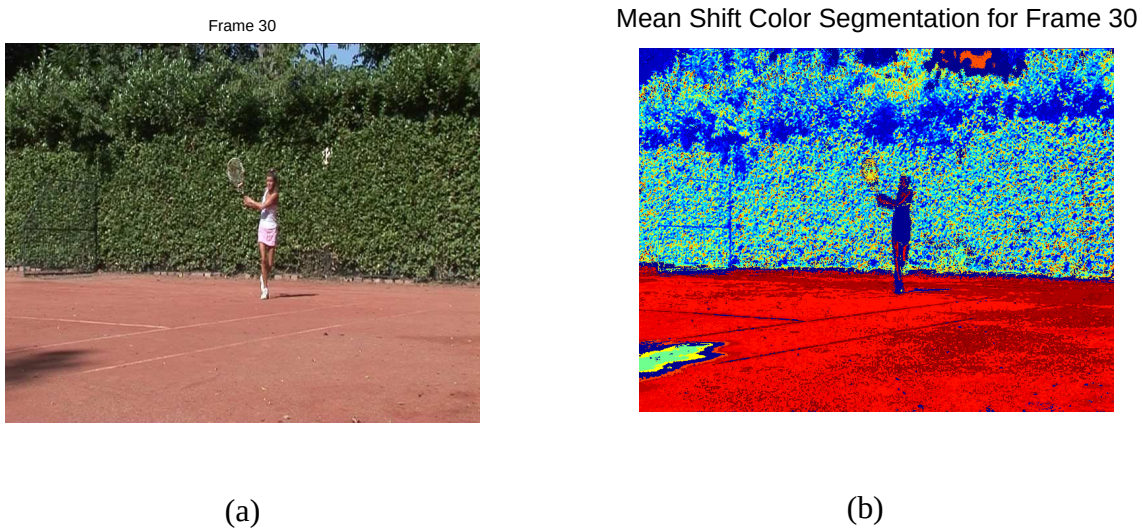

Fig. 4. Mean Shift Color Segmentation. (a) Original Frame. (b) Color Segmented Frame.

pixels are always motionless, so, by applying the mean-shift based color segmentation in those areas, we can determine which colors are present in the background. Similarly, we can separate the activity areas in color layers, corresponding to both the moving object, and the background. We then match the color segmented layers of the background to the corresponding layers in each frame's activity area, using the Earth Mover's Distance (Sec. V-A). The parts of a frame's activity area with a color that is significantly different from the color of the background are considered to belong to the moving object. This is essentially a logical 'AND' operation, where the pixels that are both in an activity area, and have a different color from the background pixels, are assigned to the moving object. The proposed method of incorporating the color information in the system has the advantage of being robust to variations in illumination and color between different video frames (or even different videos). This is because it compares the colors of different regions within a single frame, rather than between different frames, which may suffer from changes in lighting, effects of small moving elements (e.g. small leaf motions leaves in the background), or other scene arbitrary variations [45].

A. Earth Mover's Distance

Numerous techniques have been developed for the comparison of color distributions, which in our case are the color layers of the activity areas with the layers of the static frame pixels. In order

to compare color distributions, the three-dimensional color histogram can be used. However, accurately estimating the joint color distribution of each color cluster is both difficult and computationally demanding. The subsequent comparison of the three-dimensional distributions of each cluster further increases the computational cost. Additionally, in our application, the color histograms of all segmented areas, in all video frames, need to be compared, something which can easily become computationally prohibitive. Consequently, we examine the histogram of each color component separately, assuming that they are uncorrelated and independently distributed. This assumption is not true in practice, since the color channels are actually correlated with each other. Nevertheless, it is made in the present work because of computational cost concerns. In order to verify the gain in computational efficiency experimentally, we conducted experiments where the three-dimensional color histogram was used, for a short video, with only 20 frames. The color comparison took about 50 sec on a Pentium IV dual core PC for this very short video, whereas when the color channels were compared independently, the comparison took only 6.3 sec. This is due to the fact that the joint color distribution requires the computationally expensive inversion of the joint covariance matrix [45]. In practice, our experiments show that we obtain good modeling results, at a low computational cost. Naturally, examining the use of more precise color models, that are also computationally efficient, is also possible as a topic of future research.

The histograms of each color are essentially data "signatures", which characterize the data distribution. In general [48], signatures have a more general meaning than histograms, e.g. they may result from distributing the data in bins of different sizes, but we focus on the special case of color histograms. A measure of the similarity between signatures of data is the Earth Mover's Distance (EMD) [48], that calculates the cost of transforming one signature to another. A histogram with m bins, can be represented by $P = \{(\mu_1, h_1), ..., (\mu_m, h_m)\}$, where μ_i is the mean of the data in that bin, and h_i is the corresponding histogram value (essentially the probability of the values of the pixels in that cluster). This histogram can be compared with another, $Q = \{(\mu_1, h_1), ..., (\mu_n, h_n)\}$, by estimating the cost of transforming histogram P to Q. If the distance between their clusters is d_{ij} (we use the Euclidean distance here), the goal of transforming one histogram to the other is that of finding the flow f_{ij} that achieves this, while

minimizing the cost:

$$W = \sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}. \tag{7}$$

Once the optimal flow f_{ij} is found [48], the EMD becomes:

$$EMD(P, Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} d_{ij} f_{ij}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{ij}}. \tag{8}$$

We estimated the EMD between the three histograms of each color layer in the action mask and the background area of each frame. We combined the EMD's results for each color histogram by simply adding their magnitudes. The color layers of the static areas and the action areas that require the least cost (EMD) to be transformed to each other should correspond to pixels with the same color. The maximum required cost of transformation from one color signature to the other that is considered to signify similar colors was determined empirically, using the test sequences of Sec. VI as well as ten other similar real videos (as was the case in the previous sections). In our experiments, color layers that belong to the activity area and exceed this maximum cost of transformation for all color layers of the background area (of the same frame) are identified as belonging to the moving object. Our experiments show that this approach indeed correctly separates the background pixels in the action areas from the moving objects.

B. Extracted Shape Descriptors

Once the moving entities are segmented, we have a complete description of the humans that are moving in the scene under examination. Their color and their overall appearance can be used for classification, recognition (e.g. for specific tennis players or actors), categorization and in general, analysis of their actions. The shape of the moving entities captures characteristic poses during, for example, a tennis game, walking, running, and other human activities. It can also help determine which part of the activity is taking place (e.g. the player is waiting for the ball or has hit it) and can be incorporated in a system that matches known action shapes with those extracted from our algorithm. Consequently it will play a very important role in discerning between different events or classifying activities.

Fig. 5 shows three characteristic shape masks that are extracted, which essentially show the silhouette of the player. In Table II we see the MPEG-7 descriptor parameters for these "poses". Poses 2 and 3 only show the silhouette of the player, as she is standing and waiting for the

TABLE II

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF TENNIS HIT.

Frames	Global Curvature	Prototype Curvature	Smoothing Stage for Maximum Curvature
$Pose1$	(30,8)	(2,9)	48
$Pose2$	(12,18)	(1,4)	29
$Pose3$	(13,8)	(1,4)	26

ball. Both these poses differ from pose 1, where the silhouette of the racket can also be seen, as she is preparing to hit the ball. The corresponding shape descriptors reflect these similarities and differences, as the curvature zero-crossings for pose 1 are maximized after more stages than for pose 2 and 3, namely after 48 instead of 29 and 26 stages, respectively. This is because the racket's contour is more visible in the first pose, and introduces a large curve in the silhouette, which is effectively "detected" by the shape descriptor.

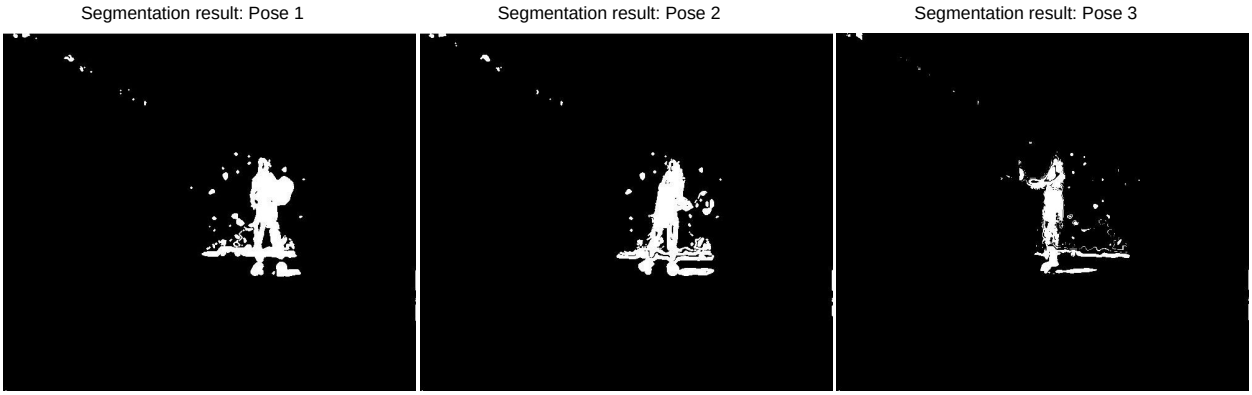

Fig. 5. Segmentation masks for different player "poses".

In many practical situations, there are many moving entities in a scene, for example in a video of a sports game with many players. In that case, the activity area, and the final segmentation results, consist of multiple connected components. These are examined separately from each other, and the shape descriptor is obtained for each one. The classification or characterization of the activity taking place is similar to that for only one moving object. There may also be many small erroneous connected components, introduced by noise. In practice, these noise-induced

regions are usually much smaller than the regions corresponding to the moving entity (e.g. in Fig. 5), so they can be eliminated based on their size. For example, in the experiments using videos of the tennis player hitting the ball or performing a tennis serve (Sec. VI-B, VI-C), morphological opening using a disk shaped structuring element of radius 2 led to the separation of the tennis ball from the player. The same sized structuring element was used in Sec. VI-A, VI-E, which contained large activity areas, whereas a radius of 1 was used in Sec. VI-D and VI-F, as the activity areas in these videos contained fewer pixels. In some cases, this leads to the "loss" of small objects, such as the tennis ball, in Sec. VI-C, but in other videos, e.g. in Sec. VI-D, small objects like the ball are retained. It should be noted that, for the particular case of tennis videos, the tennis ball is actually not present in many of the video frames. This is due to its high speed, which requires specialized cameras, in order to capture its position in each video frame. Thus, localizing and extracting it is not very meaningful in many of the sports videos used in practice.

After separating the objects in the video, the remaining connected components are then characterized using the CSS shape descriptor, which is used to categorize the activity taking place. It is very important to note at this point that, even if the smaller "noisy" connected components are not removed, they do not significantly affect the recognition rates, as they would not lead to a good match with any different activity. Similarly, when small components are lost (e.g. the tennis ball), this is very unlikely to affect recognition rates, since the smaller moving objects do not play a significant role in the recognition of the activity, which is more heavily characterized by the shape of the larger activity areas. A future area of research involves the investigation of methods for the optimal separation of the moving entities. Nevertheless, the videos used in the the current work, and the corresponding experimental results, adequately demonstrate the capabilities of the proposed system.

VI. Experiments

We applied our method to various real video sequences, containing human activities of interest, namely events that occur in tennis games and in surveillance videos. These experiments allow us to evaluate the recognition performance of our algorithm, e.g. in cases where similar activities are taking place, but are being filmed in different manners, or are being performed in different ways. The recognition performance of the proposed method is also compared against the Motion

Energy Image (MEI) method of [7], using a similar, shape descriptor-based approach, as in that work.

A. Hall sequence

In this experiment, we show the activity areas extracted for the Hall sequence (Fig. 6 (a)), where one person is entering the hallway from his office, and later another person enters the hall as well. An example of optical flow estimates, shown in Fig. 6(b), shows that the extracted velocities are high near the boundaries of the moving object (in this case the walking person), but negligible in its interior. Figs. 6(c)-(e) show the activity areas extracted for a video of the office hallway and Fig. 6(f) shows the MEI corresponding to the activity in frames $30-40$, extracted from the inter-frame differences, as in [7]. Although this is an indoor sequence with a static camera, the MEI approach leads to noisy regions where motion is supposed to have occurred, as it suffers from false alarms caused by varying illumination. It should be noted that the MEIs we extracted were obtained using the best possible threshold, based on empirical evidence (our observations), as there is no optimized way of finding it in [7]. The kurtosis-based activity areas, on the other hand, are less noisy, as they are extracted from the flow field, which provides a more reliable measure of activity than simple frame differencing. Also, the higher order statistic is more effective at detecting outliers (i.e. true motion vectors) in the flow field, than simple differencing.

Table III shows the shape parameters for activity areas extracted from subsequences of the Hall sequence. The activity areas of frames $22-25$ and $30-40$ have similar shape descriptors, with maximum curvature achieved after 45 and 51 stages respectively. This is expected, as they contain the silhouette of the first person walking in the corridor, and their main difference is the size of the activity region, rather than its contour. In frames $60-100$ there are two activity areas (Fig. 6(e)), as the second person has entered the hallway, so the shape descriptors for the activity area on the left and right are estimated separately. The parameters for these activity areas are quite different from those of Fig. 6(c), (d), because they have more irregular shapes, that represent different activities. Specifically, the person on the left is bending over, whereas the person on the right is just entering the hallway.

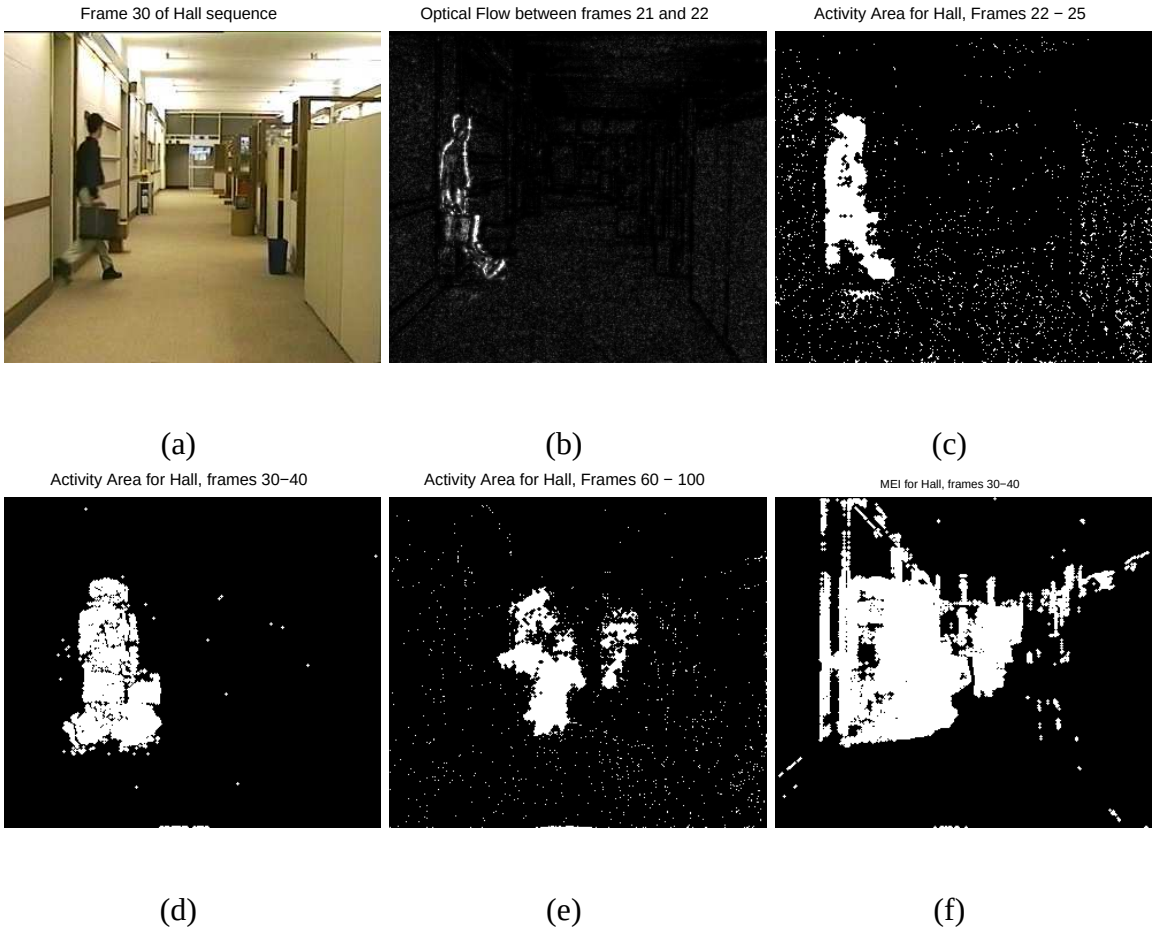

Fig. 6. Hall Sequence: (a) frame 30, (b) Optical Flow for frames $21-22$. Activity area for frames: (c) $22-25$, (d) $30-40$, (e) $60-100$. (f) MEI for frames $30-40$.

B. Tennis hit

In this video, the tennis player throws the ball in the air, then hits it, and also moves to the right to catch and hit the ball again as it returns. Frames 1 and 20 are shown in Fig. 3(a), (b), before and after the player hits the ball. The results of the optical flow between frames $9-10$ are shown in Fig. 7(a): the flow has higher values near the moving borders of the objects, but illumination variations and measurement noise have also introduced non-zero flow values in motionless pixels. Thus, by applying the method of Sec. II we remove the noise from the flow estimates and retrieve the activity areas of Fig. 3(c)-(e). These activity areas contain the pixels where the player was moving, as well as the ball trajectory, but have eliminated the effect of small background motions, such as the motion of the leaves in the background, and small motions

TABLE III

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF HALL SEQUENCE.

Frames	Smoothed Curvature	Original Curvature	Smoothing Stage for Maximum Curvature
$22-25$	(63,6)	(5,15)	45
$30-40$	(63,2)	(2,7)	51
$60-100$ left	(51,5)	(14,32)	75
$60-100$ right	(47,3)	(16,22)	67

caused by camera instability. This shows that the kurtosis is indeed robust to small motions, and can effectively separate them from true object motions. The MEI corresponding to this sequence is also extracted, and shown in Fig. 7(b). Obviously, simple frame differencing and thresholding is not sufficient for this sequence, as small motions of the leaves in the background are mistaken for a large moving area. The corresponding shape descriptors for the activity areas are shown in Table I, where the activity areas of Fig. 3(c), (d) have similar shape parameters, requiring 11 smoothing stages, whereas the activity area of Fig. 3(e) needs 38 smoothing stages. Semantics of the player's activity can be attributed to the shapes of these activity areas. Specifically, the first two activity areas have lower curvature as they contain only one curve caused by the tennis racket swinging, indicating that the player has hit the ball once. The third area contains curves on the right and left (so it needs more smoothing stages), indicating that the player hit the ball twice.

As described in Sec. IV and V, color segmentation is applied to the activity and background areas of each video frame, in order to isolate the moving entity. The results of the color segmentation are shown in Fig. 7(c), (d), where it is evident that the colors of the player and the tennis court are correctly separated. The background pixels in the activity area of a specific frame have been assigned to the same "color layer" as the corresponding areas of the background. For example, the ground in the activity area has a similar color distribution as the ground in the background. Indeed, the comparison of the color histograms using the EMD (Sec. V-A) leads to the accurate segmentation of the player in the frames, as we show in Fig. 7(e), (f).

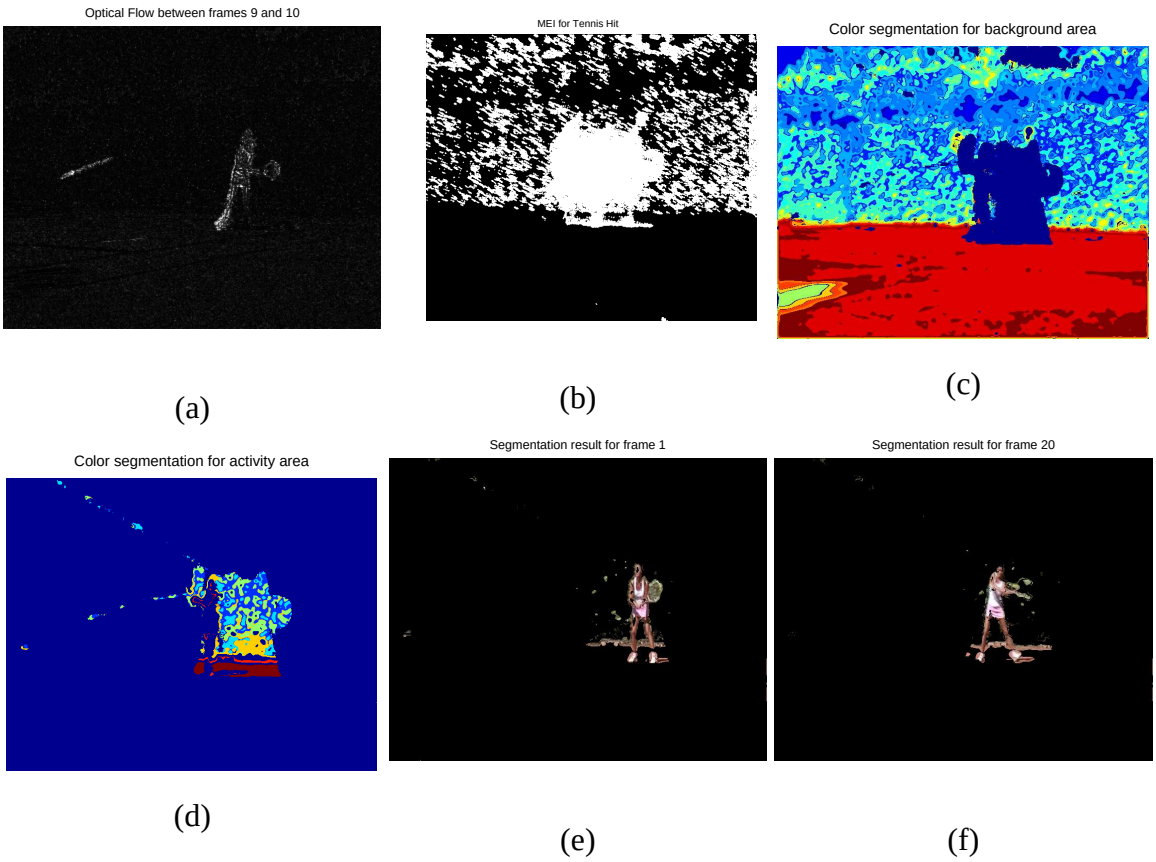

Fig. 7. (a) Optical flow, frames 9-10. (b) MEI for tennis hit. Mean Shift Color Segmentation: (c) background, (d) activity area. Segmentation results: (e) frame 1, (f) frame 20.

C. Tennis serve

This experiment uses a similar kind of video, where the player is performing a different activity, namely, serving the ball (Fig. 8 (a)). As before, the optical flow estimates do not provide sufficient information (Fig. 8 (b)). However, they lead to the activity areas of Fig. 8 (c)-(e), which are very representative of the action taking place, while eliminating the effect of small motions caused mainly by camera instability and moving leaves. On the other hand, the MEI of Fig. 8(f) is much noisier than the activity area (Fig. 8(e)), which corresponds to the same subsequence. As in the previous example, small illumination variations caused by camera instability and small leaf motions in the background create a larger activity area, whereas the actual boundaries of the player's accumulated motion are not extracted as accurately.

Fig. 8 (g) shows the results of the mean shift color segmentation applied to the activity areas.

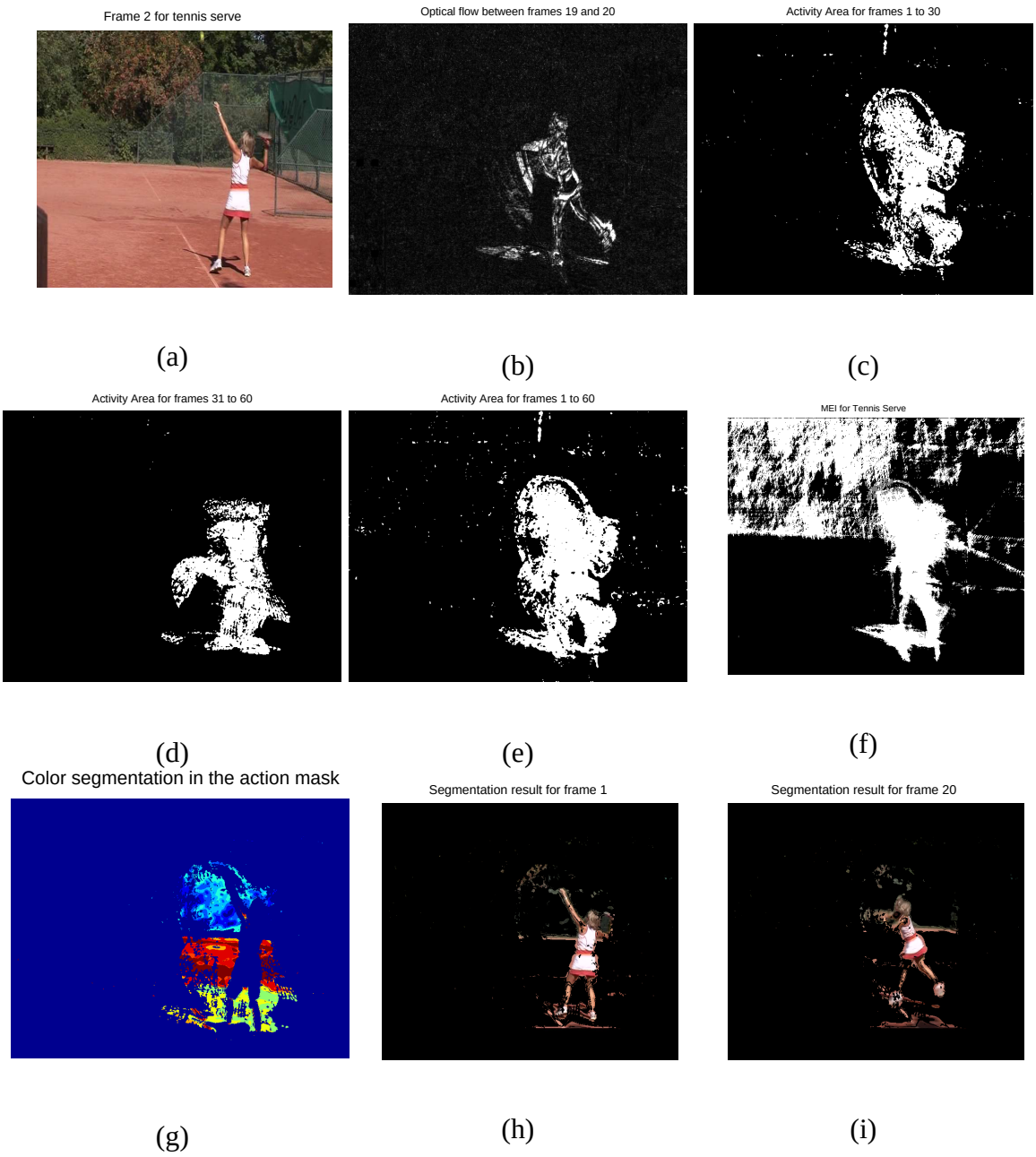

Fig. 8. Tennis serve: (a) frame 2. (b) Optical flow, frames 19-20. Activity Areas for a Tennis Serve: (c) frames $1-30$, (d) frames $31-60$, (e) frames $1-60$. (f) MEI. (g) Activity Area Color Segmentation. Segmentation results: (h) frame 2, (i) frame 20.

Once again, the color segmentation results isolate the player from her surroundings in the activity area, so we expect that matching the colors of the pixels in that area with the background colors should lead to accurate object segmentation. Fig. 8(h), (i) shows the segmentation results for

TABLE IV

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF TENNIS SERVE.

Frames	Smoothed Curvature	Original Curvature	Smoothing Stage for Maximum Curvature
$1-30$	(18,4)	(0,2)	33
$31-60$	(63,5)	(2,8)	63
$1-60$	(20,3)	(0,2)	32

two frames of this video, where it is evident that our method extracts the player with precision, despite the significant non-rigidity of her motion.

As before, we extract the MPEG-7 descriptors for this video. In Table IV, the curvature and the zero-crossing peak for the activity area of frames $1-30$ and $1-60$ are similar, which makes sense, since the second activity area includes the first. However, the activity area for frames $31-60$ has very different curvature values, and its zero-crossing maximum is achieved after 63 stages, which is expected, since that area is very jagged. Thus, the shape descriptor is a reliable measure of both how similar and how different the extracted activity areas are.

D. Tennis game

These experiments use videos of a tennis game that are different from the previous ones, as this game has been filmed from above, and shows both players hitting the ball back and forth (Fig. 9(a)). The optical flow estimates are non-negligible only near the borders of motion areas (Fig. 9(b)). This video is of very poor quality, so there are many erroneous flow estimates, caused by measurement and recording noise, as well as camera instability, slight camera panning and zoom (Fig. 9(b)). Nonetheless, the activity areas are extracted with good accuracy in Fig. 9(c)-(e): the regions where the two players are running are clearly visible, and the ball's trajectory after the serve and the successive hits by the tennis players is recovered. The activity areas for the first 50 frames (Fig. 9(c)) show that the players were running in the horizontal direction (parallel to the camera) and also contain two ball trajectories of the ball. Fig. 9(d) shows the activity area for the next 50 frames, where we see that the ball has been returned. Similarly, one can easily tell from Fig. 9(e) that over frames $20-120$ the players are moving horizontally, and that the player on the left approaches the net to return the ball. Note that the horizontal line

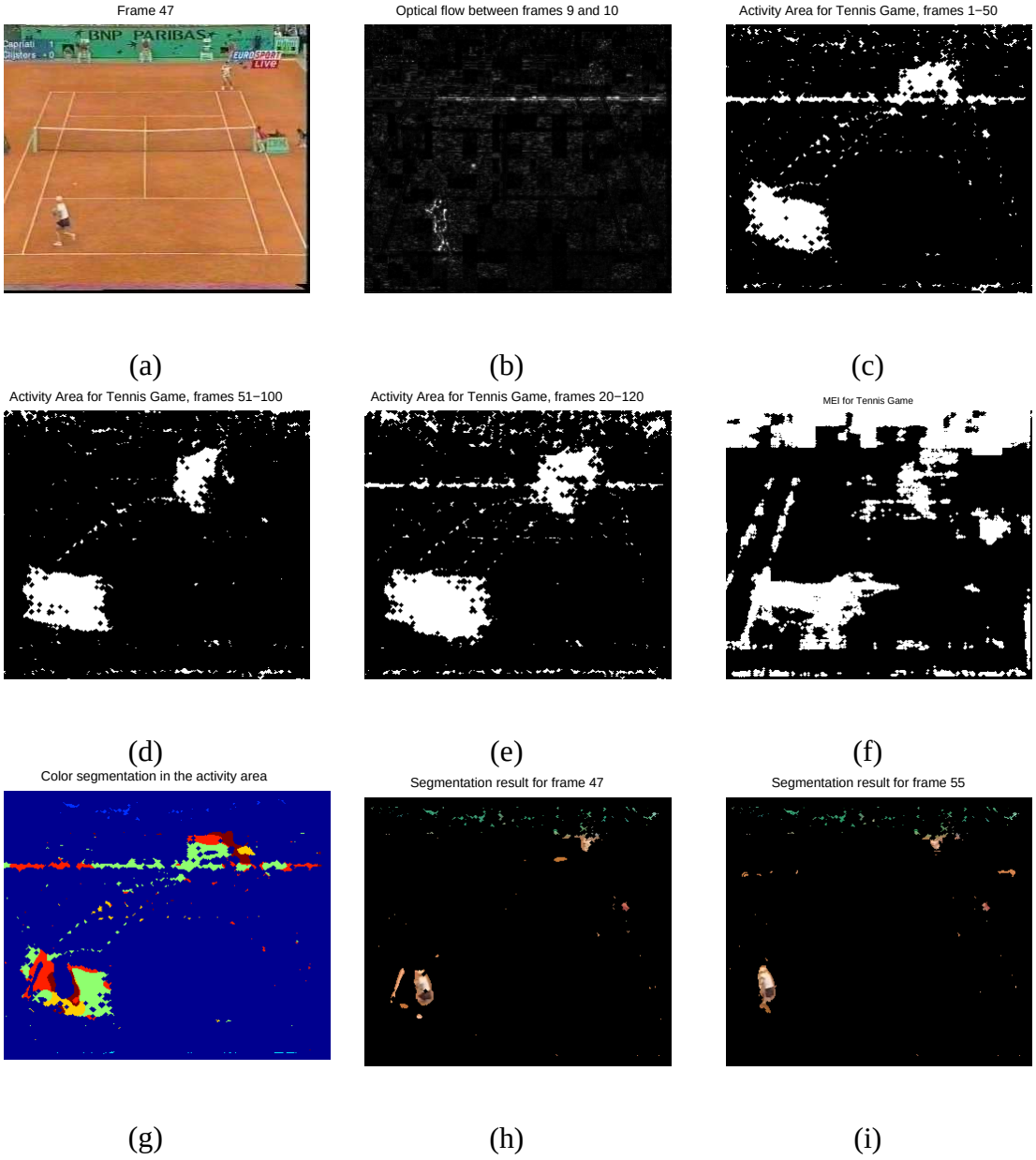

Fig. 9. Tennis game: (a) frame 47. (b) Optical flow for frames 9-10. Activity Areas for a Tennis Game: (c) frames $1-50$, (d) frames $51-100$, (e) frames $20-120$. (f) MEI. (g) Activity area color segmentation. Segmentation results: (h) frame 47, (i) frame 55.

appearing in the back is due to the small camera motions (panning, zoom) in this video. From the trajectories of the ball it is evident that the game was continuous, i.e. neither player missed the ball. Naturally, the corresponding semantics related to rules of tennis can be extracted. For the same sequence, we extract the MEI, using all frames, as in [7]. The result, shown in Fig. 9(f)

TABLE V

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF TENNIS GAME.

Event	Smoothed Curvature	Original Curvature	Smoothing Stage for Maximum Curvature
$1-50$ bottom left	(46,5)	(4,12)	45
$51-100$ bottom left	(28,8)	(4,14)	45
$20-120$ bottom left	(51,5)	(1,16)	49
$1-50$ top right	(55,7)	(6,17)	52
$51-100$ top right	(34,11)	(5,15)	51
$20-120$ top right	(53,3)	(3,20)	53

is much noisier than the activity areas extracted from the kurtosis of the optical flow. This is expected, as the simple frame differencing and thresholding cannot deal with the measurement noise, and the camera instability. Also, the MEI does not capture the tennis ball's trajectory, which is lost after thresholding the frame differences.

As before, the motion processing results are complemented by the mean shift color segmentation. In Fig. 9(g) we see the results of color segmentation on the activity areas of a frame of the video sequence. After comparing and matching the color histograms of the color layers in the activity and background areas (using the EMD), we obtain the correct segmentation of the players, shown in Fig. 9(h), (i).

We also extract the shape descriptors for the connected components in each activity area of the tennis game. Table V contains the curvature parameters for the tennis player on the bottom left and the top right, for the three subsequences with activity areas shown in Fig. 9(c)-(e). The activity areas of the bottom left player's motions are consistently smoother than those of the top right player, as the shape descriptor's values show, indicating that the top right player made more sudden motions (which is indeed the case). Also, these shape descriptors are different from those extracted for the tennis serve and tennis hit, so they can be used to classify the types of activities in tennis sequences based on their motion signatures.

E. Table Tennis

This experiment used a video of a table tennis player hitting the ball, shown in Fig. 10(a). The optical flow captures the higher velocity values, mainly at the borders of the moving objects

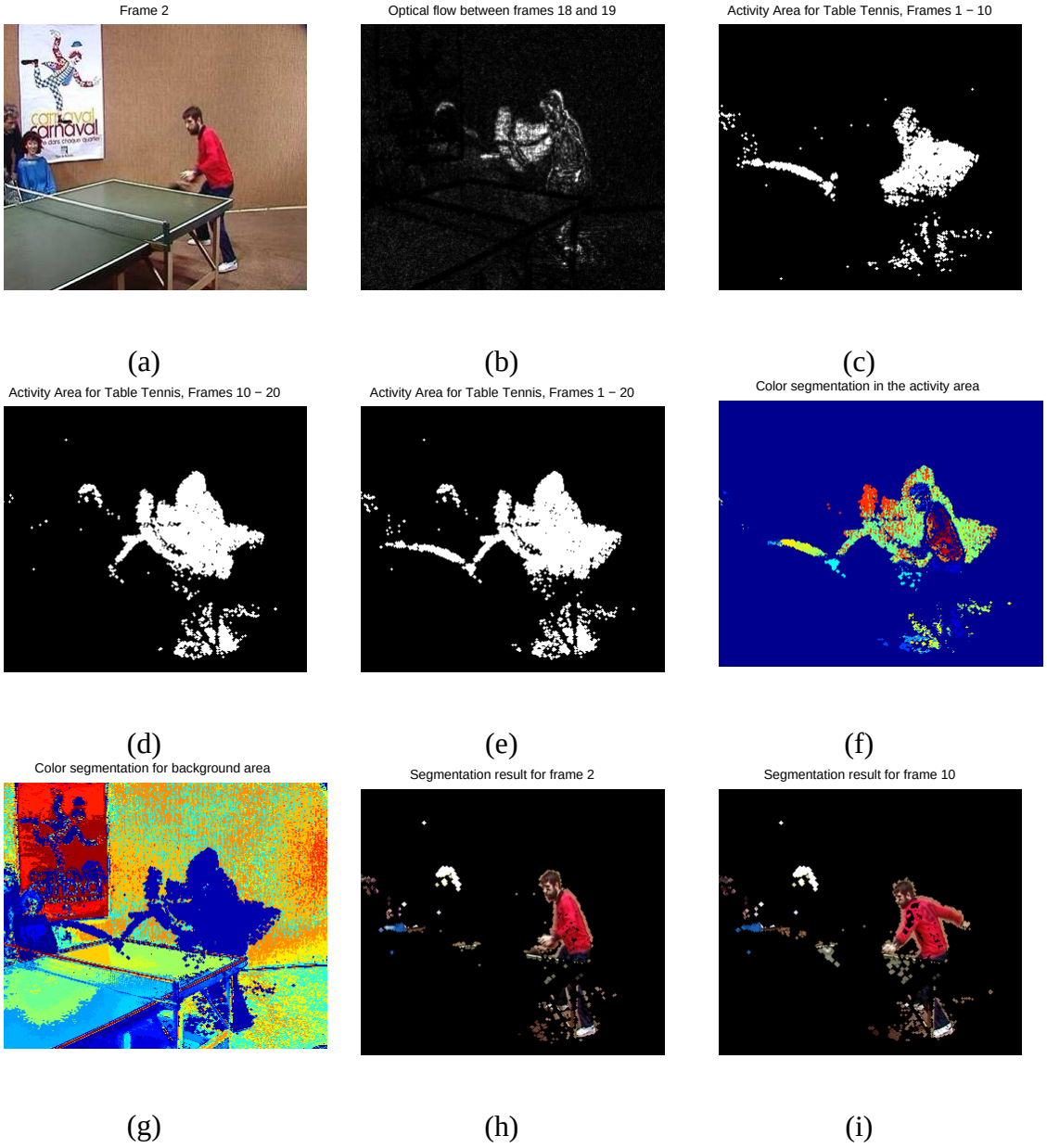

Fig. 10. Table Tennis: (a) frame 2. (b) Optical flow for Table Tennis video, frames 18-19. Activity Areas for Table Tennis: (c) frames 1-10, (d) frames 10-20, (e) frames 1-20. Mean Shift color segmentation: (f) activity area, (g) background area. Segmentation results: (h) frame 2, (i) frame 10.

(Fig. 10(b)). The corresponding activity areas are particularly characteristic of the actions taking place. In Fig. 10(c)-(e) we show the activity areas derived after accumulating the flow over the first 10 frames, the next 10, and all 20 frames together. In Fig. 10(c) the signature of the player's arm pulling back to hit the ball, and the trajectory of the approaching ball, are both clearly

TABLE VI

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF TENNIS GAME.

Frames	Smoothed Curvature	Original Curvature	Smoothing Stage for Maximum Curvature
$1-10$	(28,2)	(1,2)	35
$10-20$	(27,2)	(0,2)	31
$1-20$	(28,1)	(0,1)	30

visible. In Fig. 10 (d) the symmetric signature is extracted, that of the player hitting the ball, and the beginning of the ball's new trajectory. As expected, Fig. 10 (e) contains the union of both of these signatures, so it essentially incorporates all the motion information.

The color segmentation for the activity and background areas is shown in Fig. 10(f), (g). The player's colors are separated from the background, but they are also separated from each other (e.g. the head has different colors from the shirt). Nevertheless, the color matching procedure of section V accounts for this discrepancy, since only the pixels that match the background pixel color are removed from the activity area. The final results, shown in Fig. 10 (h), (i) show that, indeed, our method leads to very good segmentation results.

The shape descriptors for the activity areas of Fig. 10(c)-(e) are displayed in Table VI. The last two activity areas have similar shape descriptors, which is expected from the qualitative results of Fig. 10 (d), (e) as well, so these can be used to detect subsequences of similar and different activity in the table tennis video.

F. Surveillance sequence

In this experiment, surveillance videos from the PETS-CAVIAR benchmark test sequences were used to examine the performance of the system, for abnormal event detection. In this paper, we consider that a "normal" event occurs when people are only walking (or running, since it produces similar activity areas), whereas "abnormal" events can be a fight, a person falling on the floor etc. In Fig. 11(a) a sample frame from the fight sequence is shown, when the people are fighting. After the fight, one of them falls and the other runs away. The subsequence with the person running away leads to a linear activity area (Fig. 11(c)), whereas the activity area corresponding to the fight resembles a blob (Fig. 11(b)). The corresponding MEIs in Fig. 11(d),

(e) are extracted using the method of [7], by simple frame differencing and thresholding. Although this sequence is filmed from a completely static camera, indoors, the MEIs are noisier than the activity areas produced by our method, even after morphological postprocessing, due to their sensitivity to measurement noise.

In Fig. 11(f) we also show the activity area extracted from a PETS-CAVIAR sequence with only a person walking. It is obvious that the shape is linear, so blob-shaped activity areas can be interpreted as "abnormal events". The corresponding MEI, shown in Fig. 11(g), is, again, noisier than the activity area. In Fig. 11(h), (i) we also show the activity areas for a person walking and falling on the floor. Again, the activity area of the walking subsequence (Fig. 11(h)) is similar to that of Fig. 11(c), and the activity area of the abnormal event (person falling on floor) resembles a blob (Fig. 11(i)). The activity areas for these videos are combined with the color processing, as detailed in Sec. V, in order to segment out the people walking, fighting etc. Fig. 11(j), (k) show the segmentation result of people fighting (in the second of the PETS-CAVIAR "fight sequences"). It can be seen that the people are successfully segmented and that important information, related to their pose has been extracted. In Fig. 11(k) the legs of one person have not been extracted, because their color is very similar to that of the surrounding area, and barely discernible even by a human observer. The segmentation for the person walking and falling shown in Fig. 11(l) again demonstrates the effectiveness of the proposed approach.

As before, we extract the MPEG-7 descriptors for the extracted activity areas of this video. In Table VII, the curvature and the zero-crossing peak for the activity area for characteristic surveillance videos with people walking and with an abnormal event (e.g. a fight) are shown. The first line of Table VII corresponds to the activity area for the fight, in Fig. 11(b), which has smoothed curvature $(25, 2)$ and achieves maximum curvature after 35 stages. This is very different from the descriptors for the activity area corresponding to walking, in Fig. 11(c), whose smoothed curvature is $(9, 10)$ and it is achieved after 23 stages. Indeed, the curvature for another activity area corresponding to walking, in Fig. 11(h), has smoothed curvature $(8, 9)$, attained after 21 stages (the original shape was smoother than that of Fig. 11(c)). Similar results are obtained by estimating the shape descriptors for the activity areas of Fig. 11(h), (i) where a person is walking, and then falling, respectively. Table VII shows that the descriptor for the walking subsequence has smoothed curvature $(9, 9)$ and that it is attained after 22 smoothing stages. This is, again, similar to the shape descriptors for other walking sequences, but different

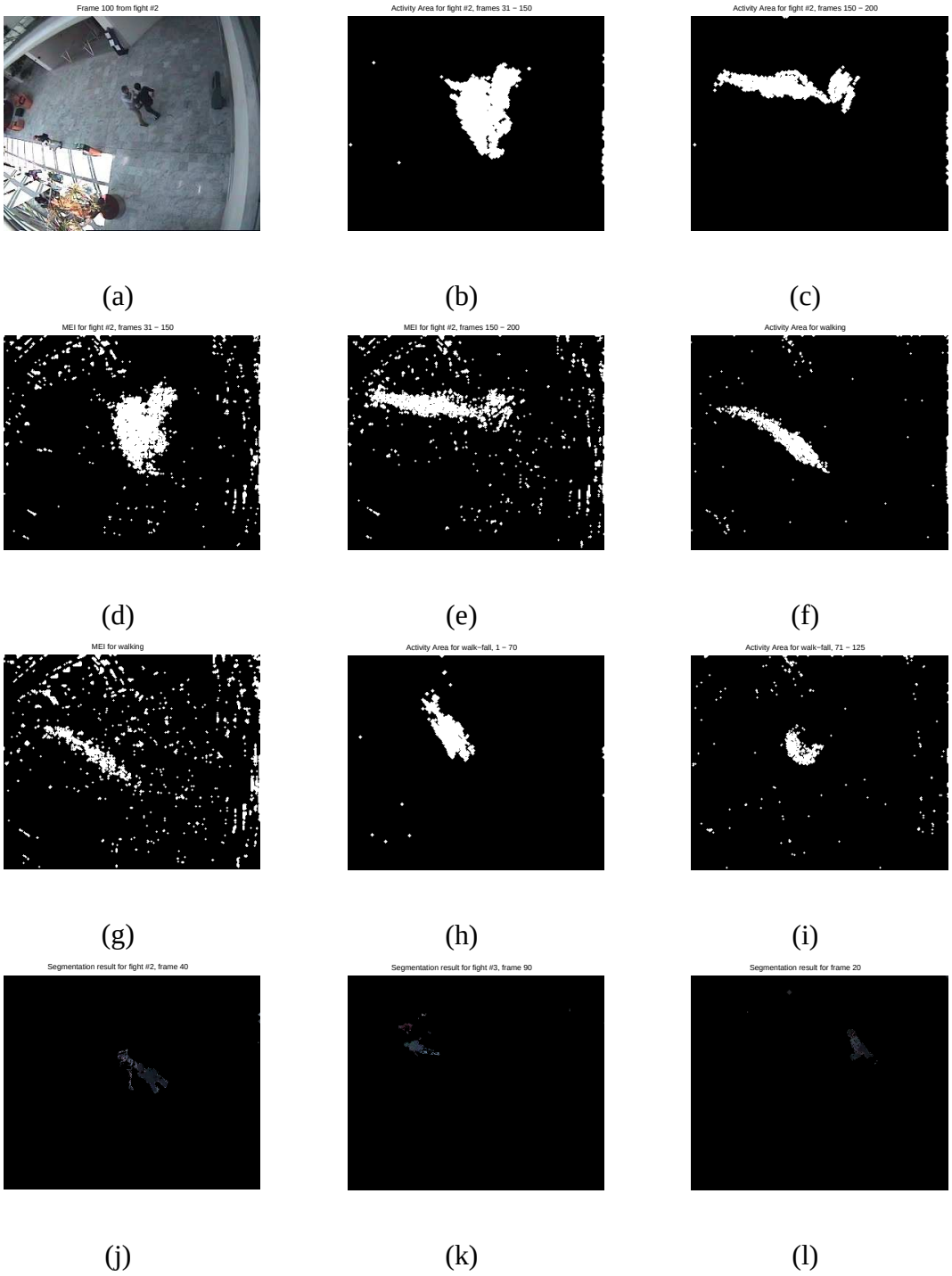

Fig. 11. Surveillance: (a) frame 100. Activity areas Fight 2: (b) fighting, (c) walking. MEIs Fight 2: (d) fighting, (e) walking. (f) Activity area for Walking. (g) MEI for Walking. Activity areas for Walk and fall: (h) walking, (i) fall. Segmentation: (j) fight 2, frame 40, (k) fight 3, frame 90, (l) walk and slump, frame 20.

TABLE VII

MPEG-7 CURVATURE DESCRIPTORS FOR THE ACTIVITY AREAS OF TENNIS SERVE.

Frames	Smoothed Curvature	Original Curvature	Smoothing Stage for Maximum Curvature
Fighting, Fig. 11(b)	(25,2)	(0,2)	35
Walking, Fig. 11(c)	(9,10)	(1,6)	23
Walking, Fig. 11(f)	(8,9)	(1,5)	21
Walking, Fig. 11(h)	(9,9)	(1,7)	22
Falling, Fig. 11(i)	(14,3)	(2,3)	33

from the descriptors for the falling down (abnormal event) activity area of Fig. 11(i), which has smoothed curvature $(14, 3)$, obtained after 33 smoothing stages.

Consequently, the smoothing stage at which the maximum curvature is attained can be used to distinguish between activity areas for walking subsequences, and "abnormal event" subsequences. This can be explained by the fact that the subsequence containing the fight has a blob-like shape that is less smooth (convex) than the shapes of the activity areas corresponding to walking and running. Thus, more stages are needed to smooth out its shape. In a similar manner, the shape of the activity areas for other subsequences, containing either walking or an abnormal event, are compared. These results are presented in Sec. VI-G, where they are also compared against the recognition performance obtained by using the MEIs of [7].

G. Recognition Performance

The shape of the extracted activity areas and their CSS shape descriptors (Sec. III-A) can be very useful for describing the human actions and, in general, the events taking place, since they become different for areas of different shape, and have similar values when the same kind of activity is occurring. We test the recognition performance of our method, as well as the performance of the Motion Energy Image (MEI) method of [7], by comparing the CSS shape descriptors corresponding to activity areas and MEIs. We examine a tennis example, where a tennis serve, a tennis hit or the tennis game itself are detected, and a surveillance application, with two people walking and fighting.

The shape descriptors extracted from walking and abnormal event (fight, fall down etc) subsequences of the surveillance videos are compared against the descriptors for walking, to

determine if a set of frames contains an abnormal event. For training, subsequences containing a walking person are extracted from the PETS-CAVIAR videos. We selected a total of 20 walking subsequences, with $100 - 300$ frames each, for which we extract the corresponding CSS descriptors, as in Table VII. As seen in Sec. VI-F, the CSS descriptors for activity areas for walking have similar values, so the mean shape descriptor for these training sequences is used to represent the activity areas that correspond to walking. For the testing, 30 subsequences with people walking are extracted from the PETS-CAVIAR videos, and 19 subsequences with "abnormal events", like fighting, falling on the floor, slumping, sitting, browsing, leaving a box on the floor, are used. For each test sequence, the activity areas are extracted, and their shape descriptors are estimated. As expected from Sec. VI-F, abnormal events lead to CSS descriptors with higher peaks than activity areas obtained from walking. We compare the descriptors from the testing sequences to the mean descriptor from the training sequences that contain walking, using the method of Sec. III-A. The proposed approach correctly identifies 75% of the abnormal events (i.e. they are not classified as walking) and 82% of the walking sequences (they are correctly classified as walking). When the same procedure takes place, using the MEIs instead of the activity areas extracted via our approach, the recognition performance falls to 68% for the abnormal events, and 70% for the walking sequences. In this set of experiments, the performance is not significantly degraded by using the MEIs, which is due to the fact that the sequences being used did not have much noise, and were filmed from a completely static camera. Consequently, the MEIs for these videos were less noisy than for the tennis videos.

For the tennis example, training takes place using a 420 sec video, with 20 subsequences showing a tennis game, that last 15 sec each, 10 subsequences with a tennis serve, of 7 sec duration, and 10 subsequences with close-ups showing a tennis hit (i.e. the player returning a serve), that last 5 sec. For testing, a 1200 sec tennis video is used, containing 38 subsequences of a tennis game that last 20 sec each, 22 subsequences of a tennis serve that last 11 sec, and 18 subsequences of a tennis hit that also last 11 sec. The entire data set also included 200 sec of subsequences with no activity of interest (no actual game): these were 14 closeups of the player taking a break, lasting 8 sec each, and 8 subsequences where the camera showed the audience for 11 sec. In these cases, an activity area which did not correspond to any "tennis-game action" (like a serve, a hit, the game) was extracted, so we refer to these as "no action sequences".

The extracted activity areas have contour shape descriptors that are sufficiently different to help

TABLE VIII

CONFUSION MATRIX FOR TENNIS GAME USING ACTIVITY AREAS.

Real event—Detected Event	Tennis Serve	Tennis Hit	Tennis Game	No Action
Tennis Serve	87.5 %	8%	2%	2.5 %
Tennis Hit	11 %	80 %	3 %	6 %
Tennis Game	1 %	1%	90 %	8 %
No Action	4 %	5%	2 %	89 %

discern between these three events, as shown in Sec. VI-B, VI-C, VI-D. Intermediate processing results for the tennis hit are shown in Sec. VI-B, for the tennis serve in Sec. VI-C, and for the tennis game itself, in Sec. VI-D. We extracted the activity areas for these activities as analyzed in Sec. II, and their corresponding MPEG-7 shape descriptors, as described in Sec. III-A. In order to compare the performance of our method with the Motion Energy Image (MEI) approach of [7], we also estimate the MEIs corresponding to the three activities of interest, and extract their shape descriptors, as in Sec. III-A.

By comparing the shape descriptors of the extracted tennis activity areas, and the respective MEIs, we obtain the detection results shown in the confusion matrix of Table VIII. The system gives overall high detection results and low “confusion rates”. The performance is better for the recognition of the tennis game, which is expected, as in this case the size of the players and the shape of the activity area is significantly different than for the close-up tennis hit and serve. Although the tennis serve and hit may be confused more easily, since they have a similar size and shape, our method achieves good recognition performance in those cases as well. The MEI-based method for the same set of activities, in the same testing and training sequences, has consistently lower recognition rates (Table IX). This is expected, since the method of [7] uses simple frame differencing to extract the MEIs, which are less accurate than the activity areas extracted by our proposed approach, as has been seen, for example, in Sec. VI-B, VI-C, VI-D. The rates of confusion are also higher when using the MEIs, as tennis serve and tennis hit produce similar MEIs, because of the noise effect. Overall, it is concluded that the activity area based method is more robust, as it can handle cases where the camera is jittery, the measurements are noisy and the illumination is varying.

TABLE IX

CONFUSION MATRIX FOR TENNIS GAME USING MEIS.

Real event—Detected Event	Tennis Serve	Tennis Hit	Tennis Game	No Action
Tennis Serve	22 %	55%	8 %	15 %
Tennis Hit	25 %	50 %	15 %	10 %
Tennis Game	13 %	60 %	12 %	15 %
No Action	11 %	9%	25 %	55 %

VII. SUMMARY AND CONCLUSIONS

In this paper we have presented a novel hybrid approach for analyzing and processing the motion and color information in a video, with the purpose of extracting information regarding the activities taking place, with emphasis on human actions. The inter-frame velocities are extracted using optical flow methods, and the resulting flow estimates are de-noised by processing their statistics. This leads to activity areas, i.e. the pixels over which there has been significant activity during the frames examined. The shape of these areas is characteristic of the actions taking place and the way the actions are being performed. MPEG-7 descriptors are extracted for the activity area contours and can be used for comparing subsequences, detecting actions and analyzing them. This information is complemented by mean shift color segmentation of the video, which provides information about the scene where the activities are taking place, and also leads to accurate object segmentation. Experiments performed with real sequences that would appear in practical applications demonstrate the usability of our approach. The proposed method is also compared against the well-known Motion Energy Images approach, and is shown to outperform it, as it is based on more robust techniques for extracting regions of activity. Finally, areas of future research include the extraction of higher level semantics by incorporating in the analysis process a priori knowledge about the video to be analyzed and logic-based processing of the resulting activity areas and masks.

ACKNOWLEDGMENTS:

This work was supported by the European Commission under contracts FP6-001765 aceMedia, FP6-027685 MESH and FP6-027026 K-Space and by the GSRT funded project DELTIO:

REFERENCES

[1] Hwang B., Kim S., and Lee S., "A full-body gesture database for automatic gesture recognition," in *Automatic Face and Gesture Recognition, 2006*, April 2006, pp. 243 – 248.

[2] Gupta L. and Suwei Ma, "Gesture-based interaction and communication: automated classification of hand gesture contours," *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, vol. 31, no. 1, pp. 114 – 120, Feb. 2001.

[3] Xu D., Yan S., Tao D., Zhang L., Li X., and Zhang H.-J., "Human gait recognition with matrix representation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 7, pp. 896 – 903, July 2006.

[4] Begg R.K., Palaniswami M., and B. Owen, "Support vector machines for automated gait classification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 52, no. 5, pp. 828 – 838, May 2005.

[5] Tovinkere V. and Qian R. J., "Detecting semantic events in soccer games: Toward a complete solution," in *Proc. IEEE Int. Conf. Mult. Expo (ICME)*, Aug. 2001.

[6] Ekin A., Tekalp M., and Mehrotra R., "Automatic soccer video analysis and summarization," *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 796 – 807, July 2003.

[7] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 257–267, Mar. 2001.

[8] Gavrila D. M., "The visual analysis of human movement: A survey," *Computer Vision and Image Understanding: CVIU*, vol. 73, no. 1, pp. 82–98, 1999.

[9] Kim S., Park C., and Lee S., "Tracking 3d human body using particle filter in moving monocular camera," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, Aug. 2006, vol. 4, pp. 805 – 808.

[10] Haiping L., Plataniotis K.N., and A.N. Venetsanopoulos, "A layered deformable model for gait analysis," in *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, April 2006, pp. 249 – 254.

[11] Chen D., Shih S., and Liao H., "Atomic human action segmentation using a spatio-temporal probabilistic framework," in *Intelligent Information Hiding and Multimedia Signal Processing, 2006. IIH-MSP '06. International Conference on*, Dec. 2006, pp. 327 – 330.

[12] Bertini M., Cucchiara R., Bimbo A., and A. Prati, "Semantic adaptation of sport videos with user-centred performance analysis," *IEEE Transactions on Multimedia*, vol. 8, no. 3, pp. 433 – 443, June 2006.

[13] Ekin A., Tekalp M., and Mehrotra R., "Automatic soccer video analysis and summarization," *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 796 – 807, July 2003.

[14] Lok W.W. and Chan K.L., "Model-based human motion analysis in monocular video," in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, March 2005, vol. 2, pp. 18–23.

[15] Laptev I. and Lindeberg T., "Space-time interest points," in *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, 2003, vol. 1, pp. 432 – 439.

[16] Oikonomopoulos A., Patras I., and M. Pantic, "Spatiotemporal salient points for visual recognition of human actions," *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, vol. 36, no. 3, pp. 710 – 719, June 2006.

[17] Yonemoto S., Nakano H., and Taniguchi R., "Real-time human figure control using tracked blobs," in *Image Analysis and Processing, 2003.Proceedings. 12th International Conference on*, Sept. 2003, pp. 127 – 132.

[18] Wren C. R., Azarbayejani A., Darrell T., and Pentland A. P., "Pfinder: Real-time tracking of the human body," vol. 19, no. 7, pp. 780–785, July 1997.

[19] McKenna S. J., Jabri S., Duricand Z., Rosenfeld A., and Wechsler H., "Tracking groups of people," *Computer Vision Image Understanding*, 2000.

[20] Thome N. and Miguet S., "A robust appearance model for tracking human motions," in *Advanced Video and Signal Based Surveillance, 2005. AVSS 2005. IEEE Conference on*, Sept. 2005, pp. 528 – 533.

[21] F. Dufaux and J. Konrad, "Efficient, robust, and fast global motion estimation for video coding," *IEEE Transactions on Image Processing*, vol. 9, no. 3, pp. 497–501, March 2000.

[22] Kanade T. Lukas B., "An iterative image registration technique with an application to stereo vision," in *International Joint Conference on Artificial Intelligence*, 1981, pp. 674–679.

[23] J. R. Berger, P. J. Burt, and K. Hanna, "Dynamic multiple-motion computation," in *Artificial Intelligence and Computer Vision: Proceedings of the Israeli Conference*, The Netherlands, 1991, pp. 147–156, Elsevier.

[24] Bouguet J., "Pyramidal implementation of the lucas kanade feature tracker description of the algorithms," in *OpenCV Documentation,Micro-Processor Research Labs, Intel Corporation*, 1999.

[25] G. D. Borshukov, G. Bozdagi, Y. Altunbasak, and A. M. Tekalp, "Motion segmentation by multistage affine classification," *IEEE Transactions on Image Processing*, vol. 6, no. 11, pp. 1591–1594, Nov. 1997.

[26] S. S. Beauchemin J. L.Barron, D. J. Fleet and T. A. Burkitt, "Performance of optical flow techniques," in *Computer Vision and Pattern Recognition, 1992. Proceedings CVPR '92, 1992 IEEE Computer Society Conference on*, June 1992, pp. 236–242.

[27] J. Weber and J. Malik, "Robust computation of optical flow in a multi-scale differential framework," *International Journal of Computer Vision*, vol. 14, pp. 67–81, Dec. 1995.

[28] H. V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, New York, second edition, 1994.

[29] G.B. Giannakis and M. K. Tsatsanis, "Time-domain tests for gaussianity and time-reversibility," *IEEE Transactions on Signal Processing*, vol. 42, no. 12, pp. 3460 – 3472, Dec. 1994.

[30] B. K. Sinha, "Detection of multivariate outliers in elliptically symmetric distributions," *The Annals of Statistics*, vol. 12, no. 4, pp. 1558–1565, Dec. 1984.

[31] J. Vogel and B. Schiele, "Semantic modeling of natural scenes for content-based image retrieval," *International Journal of Computer Vision*, vol. 72, no. 2, pp. 133157, 2007.

[32] Zibreira C. and Pereira F., "Image description and retrieval using MPEG-7 shape descriptors," in *ECDL '00: Proceedings of the 4th European Conference on Research and Advanced Technology for Digital Libraries*, London, UK, 2000, pp. 332–335, Springer-Verlag.

[33] Vretos N., Solachidis V., and Pitas I., "An MPEG-7 based description scheme for video analysis using anthropocentric video content descriptors," in *Panhellenic Conference on Informatics*, 2005, pp. 725–734.

[34] Bober M., "MPEG-7 visual shape descriptors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, pp. 716 – 719, June 2001.

[35] Mokhtarian F. and Mackworth A. K., "A theory of multiscale, curvature-based shape representation for planar curves," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 789–805, Aug. 1992.

[36] F. Mokhtarian, S. Abbasi, and J. Kittler, "Efficient and robust retrieval by shape content through curvature scale space," in *Int. Workshop on Image DataBases and Multimedia Search*, p. 35.

[37] F. Mokhtarian and A. K. Mackworth, "Matching shapes with self-intersections: Application to leaf classification," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 653–661, May 2004.

[38] Hilbert D., *Color and Color Perception*, Cambridge University Press, 1987.

[39] Goldberger J. and Greenspan H., "Context-based segmentation of image sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 3, pp. 463–468, March 2006.

[40] Comaniciu V. and Meer P., "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 5, pp. 603 – 619, May 2002.

[41] Hartigan J. A. and Wong M. A., "A K-means clustering algorithm," *Applied Statistics*, vol. 28, pp. 100–108, 1979.

[42] Fukunaga K. and Hostetler L.D., "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, pp. 32 – 40, 1975.

[43] Cheng Y., "Mean shift, mode seeking and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799.

[44] Comaniciu V. and Meer P., "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, May 2003.

[45] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Computer Vision and Pattern Recognition, 1999. Proceedings CVPR '99, 1999 IEEE Computer Society Conference on*, June 1999.

[46] A. Elgammal, R. Duraiswami, and L. S. Davis, "Efficient non-parametric adaptive color modeling using fast gauss transform," in *IEEE conference on Computer Vision and Pattern Recognition*, Dec. 2001.

[47] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction," in *6th European Conference on Computer Vision*, June/July 2000.

[48] Rubner Y., Tomasi C., and Guibas L. J., "The earth movers distance as a metric for image retrieval," *International Journal of Computer Vision*, vol. 40, no. 2, pp. 99 – 121, 2000.