Action recognition in meeting videos using head trajectories and fuzzy color histogram

Bogdan Kwolek
Rzeszów University of Technology, W. Pola 2, 35-959 Rzeszów, Poland
bkwolek@prz.rzeszow.pl

Keywords: knowledge extraction from video data, video annotation, action recognition

Received: November 30, 2004

People attending teleconference meetings usually follow specific trajectories corresponding to their intentions. In most situations the meeting video content can by sufficiently characterized by capturing the head trajectories. The tracking of the head is done using a particle filter built on cues such as color, gradient and shape. The head is represented by an ellipse with fuzzy color histogram in its interior and an intensity gradient along the ellipse boundary. By comparing pixels in entry zones to a model of the background we can detect the entry of the person quickly and reliable. The fuzzy color is constructed then in the interior of an ellipse fitting best the oval shape of the head. When a new person appears in the scene a creation of new trajectory is initialized. The recognition of actions is performed using kernel histograms built on head positions as well as segmented trajectories that are related to the layout of the room.

1 Introduction

Recent increase in the amount of multimedia data, consisting of mixed media streams, has created video retrieval an active research area. Soft computing is tolerant of imprecision, uncertainty, partial truth and provides flexible information processing ability for dealing with ambiguous situations in real-world applications. The guiding principle is to invent methodologies which lead to a robust and low cost solution of the problem. Soft computing was first proposed by Zadech [25] to construct new generation hybrid systems using neural networks, fuzzy logic, probabilistic reasoning, machine learning and derivative free optimization techniques. Soft computing based algorithms provide a very useful basis for solving many problems related to media mining. Motivated by applications, the soft computing approach has been explored by several research groups in recent years [16].

Meeting videos are important multimedia documents consisting of captured meetings in specialized smart room environments. Research activities cover for instance recording, representing and browsing of meeting videos. Speech can be very useful cue in indexing videos, but precise speech recognition in meting rooms remains a challenging task because of extensive vocabulary, different topics, speech styles and so on. The sound cue can also be used in teleconferencing scenarios to identify the speaker and to improve the tracking performance. Indexing videos using visual content is also a challenging task. On the basis of visual cues it is possible to recognize what single participants are doing throughout the meeting. An approach to knowledge extraction from such video data is described in more detail in this paper.

Human faces are peoples' identities and play important role in human action recognition. In most situations the meeting video content can by sufficiently characterized by capturing the face trajectories. In majority of the smart meeting rooms the video cameras are placed in fixed locations. The coarse extraction of foreground regions can be realized by comparing each new frame to a model of the scene background. In videos captured with fixed cameras we can distinguish several features which remain in constant geometrical relations. Taking into account the specific

structures of the meeting room we can specify head-entry and head-exit zones, which can then be utilized to detect events such as person entry and person exit. The shape of the head is one of the most easily recognizable human parts and can be reasonably well approximated by an ellipse [2]. The entry/exit events can therefore be detected when a foreground object with an elliptical shape has been found in the mentioned above zones.

The trajectories of heads have been extracted on the basis of estimates of positions produced by particle filters. The particle filters are built on cues such as color, gradient and shape. The appearance of the tracked head is represented by an ellipse with fuzzy color histogram in its interior and an intensity gradient along the ellipse boundary. The fuzzy histogram representing the tracked head has been adapted over time. This makes possible to track not only the face profile which has been shot during initialization of the tracker in the entry/exit zones but in addition different profiles of the face as well as the head can be tracked.

When fixed cameras are utilized in a meeting room we can recognize specific actions which have been performed at specific locations. Considering the fact that the location of many elements in the meeting room occupies fixed places (tables, seating, boards, microphones, etc.) we can recognize actions of participants using the declarative knowledge provided graphically by the user in advance and information provided by the visual system. The visual system yields trajectories. Each of them contains a sequence of successive head positions of the same person. This allows us to distinguish between the actions of various persons taking part in an activity.

The paper is organized as follows. After discussing related work we will present particle filtering in section 3. Then we describe the face tracking algorithm and present some tracking results. After that we demonstrate how background modeling that is based on non-parametric kernel density estimation can be used to effectively determine the person entry. In section 6 we discuss our approach to recognition of actions in meeting videos. Finally, some conclusions are drawn in the last section.

2 Related work

An overview of human motion analysis can be found in work [1]. Davis and Bobick carry out tracking of human movement using temporal templates [6]. Their method is view specific and is based on a combination of Motion Image Energy and a scalar valued Motion History Image. Yacoob and Black proposed a recognition method of activities consisting of repeated patterns [26]. The Principal Component Analysis is utilized to perform warping of the observed data to the model data. The system which has been developed by Madabhushi and Aggarwal is able to classify twelve different classes of actions [15]. These actions are walking, standing up, sitting, getting up, bending, bending sideways, falling, squatting, rising and hugging in the lateral or frontal views. Each test sequence was a discrete action primitive. A recognition rate about of 80 percent has been achieved. In the area of action and activity recognition the Hidden Markov Models [20] are widely used by several research groups [12][17]. The HMMs require a large amount of training data in the spatio-temporal domain for actions and events to be recognized. The most of the existing approaches require either large training data for recognition of actions at acceptable level, or a specific number of people for training the system. A retraining of the system which requires a large amount of video data might not be feasible in several real-world situations.

3 Generic particle filtering

Applying face detection procedure to each frame during video content analysis can be inefficient because of significant computational load. The variation of a face within a continuous shot is typically small. Taking into account the continuity between consecutive frames, the tracking algorithms conduct searching only in a reduced area in the neighborhood of the face found according to the model constructed in the earlier frame for the corresponding face, instead the processing the whole image. The models are typically updated frame by frame to reflect object changes over time.

In soft belief systems a weight is attached to each hypothesis. The degree of a belief can be expressed via conditional probability, Dempster-

Shafer belief function or frequency of data [16]. Recently, sequential Monte Carlo methods [7], also known as particle filters, have become increasingly popular stochastic approaches for approximating posterior distributions [11] [14] [19] [24]. Particle filter operates by approximating the posterior distribution using a collection of weighted samples $C = \{X_t^{(a)}, \pi_t^{(a)}\}_{a=1}^{K}$, where each sample $X_t^{(a)}$ represents hypothesized state of the target and the weights are normalized such that $\sum_a \pi_t^{(a)}$.

The problem of tracking can be formulated as the Bayesian filtering

$$p(X_t \mid Z_{1:t}) \propto p(Z_t \mid X_t) \int p(X_t \mid X_{t-1}) \\ p(X_{t-1} \mid Z_{1:t-1}) dX_{t-1} \qquad (1)$$

where X_t and Z_t denote the hidden state of the object of interest and observation vector at discrete time t, respectively, whereas $Z_{1:t} = \{Z_1, ..., Z_t\}$ denotes all the observations up to current time step. With this recursion we can calculate the posterior $p(X_t \mid Z_{1:t})$, given a dynamic model $p(X_t \mid X_{t-1})$ describing the state propagation and an observation model $p(Z_t \mid X_t)$ describing the likelihood that a state X_t causes the measurement Z_t together with the following conditional independence assumptions: $X_t \perp Z_{1:t-1} \mid X_{t-1}$, $Z_t \perp Z_{1:t-1} \mid X_t$.

The evolution of the sample set takes place by drawing new samples from a suitably chosen proposal distribution which may depend on the old state and the new measurements, i.e. $X_t^{(a)} \sim q\Big(X_t \mid X_{t-1}^{(a)}, Z_t\Big)$ and then propagating each sample according to probabilistic motion model of the target. To give a particle representation of the posterior density the samples are set to $\pi_t^{(a)} \propto \pi_{t-1}^{(a)} p\Big(Z_t \mid X_t^{(a)}\Big)$ $p\Big(X_t^{(a)} \mid X_{t-1}^{(a)}\Big) / q\Big(X_t^{(a)} \mid X_{t-1}^{(a)}, Z_t\Big)$.

The particles should be re-sampled according to their weights to avoid degeneracy. Particle filters rely on importance sampling and in consequence their performance depends on the nature of the proposal distribution. To implement the particle filter one needs to know the initial condition $p(X_0 \mid Z_0)$, the motion model $p(X_t \mid X_{t-1})$ and the observation model $p(Z_t \mid X_t)$. The next section presents the ingredients of the particle filter.

4 Face tracking

In this section we demonstrate our tracking approach to extract face/head trajectories. We describe below the state space and the dynamical model. Next, we discuss the extraction of fuzzy color histogram using the fuzzy c-means clustering. The observation model is discussed after that. In this part we explain also how multiple cues are integrated in a probabilistic manner and describe model update over time. In the last subsection we demonstrate some tracking results which have been obtained on PETS-ICVS 2003 data sets.

4.1 State space and the dynamical model

The outline of the head is modeled in the 2D image domain as a vertical ellipse that is allowed to translate and scale subject to a dynamical model. Each sample represents a state of an ellipse that is parameterized by $X = \{x, \dot{x}, y, \dot{y}, s_y, \dot{s}_y\}$, where x and y denote centroid of the ellipse, $\dot{x}$ and $\dot{y}$ are the velocities of the centroid, s_y is the length of the minor axis of the ellipse with an assumed fixed aspect ratio and $\dot{s}_y$ is the velocity of s_y.

The samples are propagated on the basis of a dynamic model $X_t = AX_{t-1} + W_t$, where A denotes a deterministic component describing a constant velocity movement and W_t is a multivariate Gaussian random variable. The diffusion component represents uncertainty in prediction and thus provides the algorithm with a local search about the state.

4.2 Fuzzy color histogram

Digital images are mappings of natural scenes and thus possess a reasonable amount of uncertainty due to sampling and quantization [13]. A conventional color histogram considers no color similarity across the miscellaneous bins [10]. By considering inter-color distance we can construct a fuzzy color histogram [13] and thus to incorporate the uncertainty and the imprecise nature of color components. In such a histogram a pixel of a given color contributes not only to its specific bin but also to the neighboring bins of the histogram.

A color histogram can be used to represent the color distribution [21]. For an image I contain-

ing N pixels a histogram representation $H(I) = \{h_1, h_2, ..., h_n\}$, where $h_i = N_i/N$ denotes the probability that a pixel belongs to a i-th color bin, can be extracted by counting the number N_i of pixels belonging to each color bin. The probability h_i can be computed as follows [10]:

$$h_i = \sum_{j=1}^{N} P_{i|j} P_j = \frac{1}{N} \sum_{j=1}^{N} P_{i|j} \tag{2}$$

where P_j is the probability of a pixel from image I being the j-th pixel, $P_{i|j}$ is the conditional probability and it is equal to 1 if the j-th pixel is quantized into the i-th color bin, 0 otherwise. Therefore, the probability h_i can be computed on the basis of the following equation

$$h_i = \frac{1}{N} \sum_{j=1}^{N} \delta(g(j) - i) \tag{3}$$

where the function $g()$ maps the color of pixel j to bin number, and δ is the Dirac impulse function.

The value of each bin in a fuzzy histogram should represent a typicality of the color within the image rather than its probability. The fuzzy color histogram of image I can be expressed as $F(I) = \{f_1, f_2, ...f_n\}$, where the probability f_i expressing color typicality is computed as follows:

$$f_i = \sum_{j=1}^{N} \mu_{ij} P_j = \frac{1}{N} \sum_{j=1}^{N} \mu_{ij} \tag{4}$$

and μ_{ij} is the membership value of the color of j-th pixel in the i-th color bin. In order to compute the fuzzy color histogram of an image, we need to consider the membership values with respect to all color bins. The probability f_i can be expressed as follows:

$$f_i = \frac{1}{N} \sum_{j \in C} h(g(j)) \mu_{ij} \tag{5}$$

where C is the set of colors of the image I, and the function $g()$ maps the color to bin number. This equation is the linear convolution between the conventional color histogram and the filtering kernel. The convolution provides a smoothing of the histogram. This means that each pixel's color influences all the histogram bins. In work [13] such a smoothing based approach, where the influence from neighboring bins is expressed by triangular membership functions, has been used to extract fuzzy histograms of gray images.

To precisely quantify the perceptual color similarity between two colors a perceptually uniform color space should be utilized. In a perceptually uniform color space the perceived color differences recognized as equal by the human eye should correspond to equal Euclidean distances [18]. The CIELab color space [18], one of the perceptually uniform color spaces, has been utilized in this work to construct the fuzzy histogram.

The L^*, a^* and b^* components are given by:

$$\begin{aligned} L^* &= 116 g\left(\frac{Y}{Y_0}\right) - 16 \\ a^* &= 500 \left[g\left(\frac{X}{X_0}\right) - g\left(\frac{Y}{Y_0}\right) \right] \\ b^* &= 200 \left[g\left(\frac{Y}{Y_0}\right) - g\left(\frac{Z}{Z_0}\right) \right] \end{aligned} \tag{6}$$

where

$$g(x) = \begin{cases} x^{\frac{1}{3}} & x > 0.008856 \\ 7.887x + \frac{16}{116} & \text{otherwise} \end{cases}$$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4125 & 0.3576 & 0.1804 \\ 0.2127 & 0.7152 & 0.0722 \\ 0.0193 & 0.1192 & 0.9502 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

and X_0, Y_0, Z_0 represents reference white point that is determined for $[R\ G\ B]^T = [1\ 1\ 1]^T$.

In work [10] an efficient method to compute the membership values without a direct use of the color space transformation RGB $\rightarrow$ CIElab has been proposed. The membership values are computed using fuzzy c-means (FCM) algorithm [4]. The main idea of this approach is to compute in an off-line phase the membership matrix and then use it on-line to compute the membership values on the basis of colors in RGB space.

At the beginning a fine and uniform quantization consisting in mapping all colors from RGB space to n' histogram bins is performed [10]. Then, the transformation of n' bins into CIELab color space is conducted. Finally, the n' colors from CIELab space are classified to $n \ll n'$ clusters using FCM clustering technique. As a result a membership matrix $U = [u_{ik}]_{n \times n'}$ is computed. It can be then utilized on-line to compute n-bin fuzzy color histogram using the n'-bin typical histogram of the image I. The equation expressing this conversion has the following form

$$F_{n \times 1} = U_{n \times n'} H_{n' \times 1}. \tag{7}$$

In the classical k-means algorithm, each data point is assumed to be in exactly one cluster. In FCM algorithm each sample has a membership in a cluster and the memberships are equivalent to probabilities. The FCM algorithm seeks a minimum of a heuristic global cost function, which is the weighted sum of squared errors within each cluster, and is defined as follows:

$$J_{fuz}(U,v) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m \left\|x_k - v_i\right\|^2 \quad (8)$$

where U is a fuzzy c partition of the data set $X = \{x_1, x_2, ..., x_n\}$, the vector v is defined as $v = \{v_1, ..., v_2, ..., v_c\}$, and v_i is the cluster center of class i, m is a free parameter selected to adjust the extent of membership shared by c clusters. For $m > 0$ the criterion allows each data point to belong to multiple clusters. The term u_{ik} is the membership value reflecting that the individual k-th data point is in the i-th fuzzy set. The probabilities of cluster membership are normalized as $\sum_{i=1}^{n} u_{ik} = 1$, where $1 \leq k \leq n$, $u_{ik} \in [0,1]$, and $0 < \sum_{k=1}^{n} u_{ik} < n$ for $1 \leq i \leq c$. The J_{fuz} criterion is minimized when the cluster centers v_i are in proximity of those points that have high estimated probability of being in cluster i.

The cluster means and probabilities have been estimated iteratively using the following equations [4]:

$$\begin{aligned} v_i &= \frac{\sum_{k=1}^{n}(u_{ik})^m x_k}{\sum_{k=1}^{n}(u_{ik})^m} \\ u_{ik} &= \frac{1}{\sum_{j=1}^{c}\left(\frac{\left\|x_k - v_i\right\|^2}{\left\|x_k - v_j\right\|^2}\right)^{\frac{1}{m-1}}} \end{aligned} \quad (9)$$

where $1 \leq i \leq c$, and $1 \leq k \leq n$. No guarantee ensures that FCM converges to an optimum solution. The following convergence test has been utilized in each iteration l

$$\left\|U^{l-1} - U^l\right\| = max_{i,k}\left\{\left|u_{ik}^{(l-1)} - u_{ik}^{(l)}\right|\right\} < \epsilon. \quad (10)$$

The performance of FCM depends on initial clusters. In our implementation we utilized n'=512 bins in the typical histogram and n=32 bins in the fuzzy histogram.

4.3 The observation model

To compare the fuzzy histogram Q representing the tracked face to each individual fuzzy histogram F, which has been computed in the interior of the ellipse determined in advance on the basis of the state hold in the considered particle, we utilized the metric $\sqrt{1-\rho(F,Q)}$ [3]. This metric is derived from Bhattacharyya coefficient $\rho(F,Q) = \sum_{u=1}^{n}\sqrt{F^{(u)}Q^{(u)}}$. Using this coefficient we utilized the following color observation model $p(Z^C \mid X) = (\sqrt{2\pi}\sigma)^{-1}e^{-\frac{1-\rho}{2\sigma^2}}$. Applying such Gaussian weighting we favor head candidates whose color distributions are similar to the distribution of the tracked head.

The second ingredient of the observation model reflecting the edge strength along the elliptical head boundary has been weighted in a similar fashion $p(Z^G \mid X) = (\sqrt{2\pi}\sigma)^{-1}e^{-\frac{1-\phi_g}{2\sigma^2}}$, where ϕ_g denotes the normalized gradient along the ellipse's boundary. To compute the gradients and the histograms fast we prepared and stored for the future use two lists. For each possible length of the minor axis the lists contain coordinates of the outline in relation to the center as well as corresponding coordinates of all interior pixels.

The aim of probabilistic multi-cue integration is to enhance visual cues that are more reliable in the current context and to suppress less reliable cues. The correlation between location, edge and color of an object even if exist is rather weak. Assuming that the measurements are conditionally independent given the state we obtain the equation $p(Z_t \mid X_t) = p(Z_t^G \mid X_t) \cdot p(Z_t^C \mid X_t)$, which allows us to accomplish the probabilistic integration of cues. To achieve this we calculate at each time t the L2 norm based distances $D_t^{(j)}$, between the individual cue's centroids and the centroid obtained by integrating the likelihood from utilized cues [22]. The reliability factors of the cues $\alpha_t^{(j)}$ are then calculated on the basis of the following leaking integrator $\xi\dot{\alpha}_t^{(j)} = \eta_t^{(j)} - \alpha_t^{(j)}$, where ξ denotes a factor that determines the adaptation rate and $\eta_t^{(j)} = 0.5 * (\tanh(-eD_t^{(j)}) + w)$. In the experiments we set $e = 0.3$ and $w = 3$. Using the reliability factors the observation likelihood has been determined as follows:

$$p(Z_t \mid X_t) = [p(Z_t^G \mid X_t)]^{\alpha_t^{(1)}} \cdot [p(Z_t^C \mid X_t)]^{\alpha_t^{(2)}} \quad (11)$$

where $0 \leq \alpha_t^{(j)} \leq 1$.

To deal with profiles of the face the histogram representing the tracked head has been updated over time. This makes possible to track not only a

face profile which has been shot during initialization of the tracker but in addition different profiles of the face as well as the head can be tracked. Using only pixels from the ellipse's interior, a new fuzzy color histogram is computed and combined with the previous model in the following manner $Q_t^{(u)} = (1-\gamma)Q_{t-1}^{(u)} + \gamma F_t^{(u)}$, where γ is an accommodation rate, F_t denotes the histogram of the interior of the ellipse representing the estimated state, Q_{t-1} is the model histogram representing the head in the previous frame, whereas $u = 1, ..., n$.

4.4 Tracking results

The experiments described in this subsection have been realized on the basis of PETS-ICVS data sets. The images of size 720x576 have been converted to size of 320x240 by subsampling (consisting in selecting odd pixels in only odd lines) and bicubic based image scaling. The PETS data set contains several videos. For cameras 1 and 2 in scenario C there are a maximum of 3 people sitting in front of each camera. Figure 1 depicts some tracking results. The experiments have been conducted using a relatively large range of the axis lengths, namely from 6 to 30. A typical length of the ellipse's axis which is needed to approximate the heads in the PETS-ICVS data sets varies between 10 and 14. The frame-rate of the tracking module is about 12-15 Hz on a 2.4 GHz PC.

Figure 1: Tracking the face. Frame #10686 (a). Frame #14840 (b).

The related tracker [19] also uses color distributions and particle filtering for multiple object tracking. It employs typical color histogram while we use fuzzy histogram. By employing fuzzy histogram our tracker can track objects more reliably in cases of illumination changes and temporal occlusions. The methods differ in the model update, shape representation and initialization of the tracker. The initialization of the tracker is discussed in the next section.

5 Background subtraction using a non-parametric model of the scene

In most of the smart meeting rooms the video cameras are placed in fixed locations. The camera locations should be chosen carefully to capture the meetings with little occlusions as possible. The lighting conditions should provide the repetitive appearance of objects during realization of particular actions. In meeting scenarios the detection of foreground regions can be realized by comparing each new frame to a model of the scene background. Since person actions are always coupled with motion, our approach utilizes the model of scene background to detect the person entry/exit events. A background subtraction technique is used to initialize the tracker as well as to provide the tracker with additional information about possible locations of objects of interests. The initialization of the tracker has been performed by searching for an elliptical object in determined in advance head-entry and head-exit zones. A background subtraction procedure which was executed in mentioned above boxes has proven to be sufficient in detection of person entry.

In work [9] the background of an image is extracted on the basis of collection of pixels considered as being the background in a sequence of images. The robust background extraction is based on estimation of density function of the density distribution given a history of pixel values. The model of the background holds a sample of intensity values for each pixel in the image and uses this sample to estimate the probability density function of the pixel value. If $S = \{x_1, x_2, ..., x_L\}$ is a recent sample of intensity values for a gray pixel, the probability density function that this pixel will have intensity value x_t at time t can be non-parametrically estimated using the kernel K_h as $Pr(x_t) = \frac{1}{L}\sum_{i=1}^{L} K_h(x_t - x_i)$. For Gaussian kernel $K_h = N(0, \Sigma)$ and a given sample $S = \{x_i\}_{i=1}^{L}$ from a distribution with density $p(x)$, where $\Sigma = \sigma^2$ represents the kernel bandwidth, an estimate of this density at x can be calculated as follows [9]:

$$Pr(x) = \frac{1}{L}\sum_{i=1}^{L} \frac{1}{\sqrt{2\pi\sigma^2}} \exp -\frac{1}{2}\frac{(x-x_i)^2}{\sigma^2} \qquad (12)$$

The pixel is considered as a foreground if $Pr(x) < threshold$. The kernel bandwidth expresses the local variation in the pixel intensity due to image blur and not the intensity jumps. The local variance varies over the image and changes over time. The standard deviation was estimated using the following equation [9]:

$$\sigma = \frac{1}{0.68\sqrt{2}(L-1)} \sum_{i=1}^{L-1} |x_i - x_{i+1}|. \qquad (13)$$

Figure 2 demonstrates exemplary result of background subtraction, which has been obtained for $L = 10$. The threshold has been set to 0.1. The probabilities have been calculated using precalculated lookup tables for the kernel function.

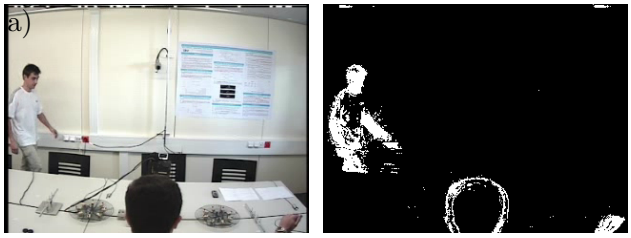

Figure 2: Frame #10683 of Scenario C viewed from Camera 1 (a). Background subtraction (b).

6 Action recognition

In the first subsection of this section we demonstrate the framework for action recognition relying on spatial relations of objects as well as domain knowledge. In the next subsection we discuss the segmentation algorithm of video streams. The section explains also how actions are recognized using prior isolated sequences of action features.

6.1 Action recognition using spatial relations

In a meeting room there are typical locations, where the participants perform particularly interesting activities, such as conference tables, whiteboards, projection screens, and where the actions should be recognized more perfectly. In meeting videos which are captured with fixed cameras it is possible to distinguish specific structures depending on world constraints. The locations of many elements in the meeting room remain fixed. Therefore the visual structure of the images varies very little over multiple meetings. Such scene structures remaining within mutual context can be used in an automatic recognition of simple individual or group actions and events. The recognition can be realized using absolute and relative positions between objects and heads.

The module works on the basis of head locations coming from the tracking module and the knowledge provided in advance by the user. The recognition rules are generated on the basis of rectangular zones specified with a graphical user interface. The zones are used to define the specific actions at particular places. The drawing tool allows the user to easily create the spatio-temporal action templates. The location of a template can be absolute or relative. Each zone can be in state *on* or *off*. A zone is in state *on* when a head is currently inside the specified area. It is possible to join the rectangular zones using arrows. For the absolute zones the axes are used to specify possible paths or trajectories of the head. In case of the relative boxes the axes can be used to specify spatial relations. To define trajectories the user can specify a time-line separately for axes and boxes. 3-4 zones usually specify a typical trajectory. Thanks to keeping the consecutive positions of particular heads the recognition module can take into account the temporal locations of objects of interest (movement and duration of presence). The trajectories allow us to distinguish between the actions of various persons taking part in an activity. This approach has proved particularly useful in recognizing actions in PETS-ICVS data sets because the available training material is too limited. A disadvantage of the drawing tool in its present version is that it can only be used with static images.

6.2 Segmentation of video streams using the Bayesian Information Criterion

The Bayesian Information Criterion (BIC) as the model selection criterion has been used in [5]. The problem of model selection consists in selecting one among a set of candidate models in order to represent a given data set. In the mentioned above work the segmentation/clustering problem has been formulated as the model selection between two nested competing models on the basis

of comparison of BIC values. Several desirable properties of the method, such as threshold independence, optimality and robustness have been demonstrated as well. In recent years BIC has been mainly used in speech systems in segmentation and segments clustering. In this work the temporal segmentation of streams consisting of feature sequences has been realized on the basis of an efficient variant of BIC introduced by Tritschler and Gopinath [23]. In order to improve the precision, especially on small segments, a new windows choosing scheme has been proposed.

Denote $X = \{x_i\}_{i=1}^{M}$ where $x_i \in R^d$ as the sequence of frame-based feature vectors extracted from a video stream in which there is at most one segment boundary. Our intention is to determine all possible frames where there is a boundary segment. If we suppose that each feature block can be modeled as one multivariate Gaussian process, the segmentation can be treated as a model selection problem between the following two nested models [5][23]: model Q_1 where $X = \{x_i\}_{i=1}^{M}$ is identically distributed to a single Gaussian $N(\mu, \Sigma)$, and model Q_2 where $X = \{x_i\}_{i=1}^{M}$ is drawn from two Gaussians while $\{x_i\}_{i=1}^{b}$ is drawn from one Gaussian $N(\mu_1, \Sigma_1)$, and $\{x_i\}_{i=b+1}^{M}$ is drawn from another Gaussian $N(\mu_2, \Sigma_2)$. Since $x_i \in R^d$, the model $Q1$ has $k_1 = d + 0.5d(d+1)$ parameters, while the second model Q_2 has twice as many parameters. The b-th frame is a good candidate for the segment boundary if the BIC difference

$$\begin{aligned} \Delta BIC_b &= \tfrac{1}{2} M \log|\Sigma| - b \log|\Sigma_1| \\ &\quad -(M-b)\log|\Sigma_2| \\ &\quad -\tfrac{1}{2}\lambda\left(d + \tfrac{1}{2}d(d+1)\right)\log M \end{aligned} \tag{14}$$

is negative, where $|\,|$ denotes the matrix determinant, Σ is the covariance matrix of the whole stream consisting of M samples, Σ_1 is the covariance of the first subdivision, Σ_2 is the covariance of the second subdivision, and λ is penalty weight. The BIC difference can be seen as an approximation of the logarithm of the Bayes factor. The final segmentation decision can be obtained via MLE and applying this test for all possible values of b and choosing the most negative ΔBIC_b, $\hat{b} = \arg\max_b \Delta BIC_b$. If no segment boundary has been found on the current window, the size of the window is increased [23].

The experiments have shown that good segmentation results can be obtained using the energy cue. The initial window length with 15 features gives optimal segmentation results. Figure 3 illustrates exemplifying segmentation results which were obtained for Person 2 in PETS-ICVS data sets (scenario C, camera 1, person sitting in the middle, see Fig. 1b).

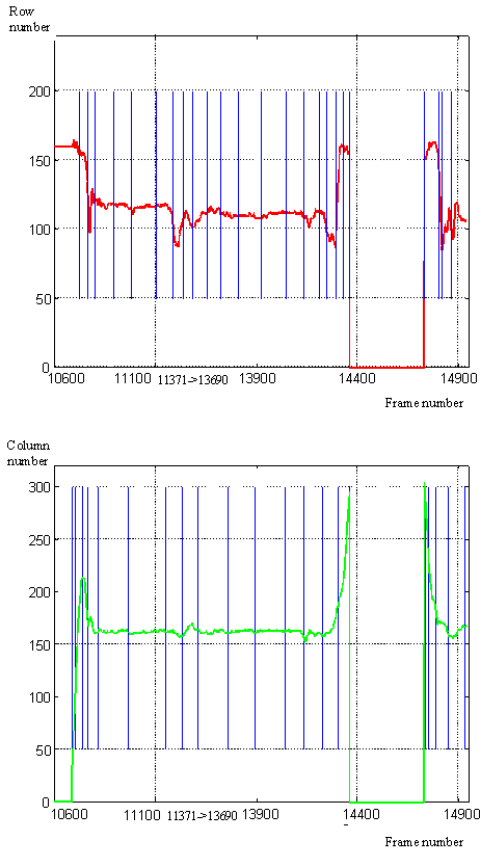

Figure 3: Segmentation of temporal trajectories.

7 Experiments

The trajectory of Person 2 that has been obtained using the images acquired by Cam1 and the face/head tracker is depicted in Fig. 4. The performance of the recognition module has been evaluated on a part of the PETS-ICVS data set (scenario A and C, camera 1 and 2).

On the basis of coherency in time and space between indexes generated by the spatio-temporal recognizer and the BIC based segmentation of trajectory we extracted the segments consisting of head positions. The histograms reflecting executed actions have been constructed using a Gaussian kernel [8][14]. During extraction of a his-

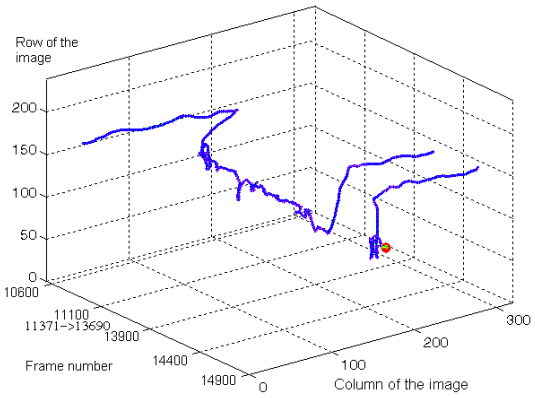

Figure 4: Trajectory of Person 2 in PETS-ICVS data sets, Scenario C, Cam1.

togram the kernel has been utilized to weight the head coordinates according to their distance to the center of the kernel. The larger the distance of the head from the kernel center, the smaller the weight. The kernel center has been located at the last position in an extracted segment. Figure 5 illustrates exemplar histograms which have been obtained from two different scenarios A and C. The first histogram from this figure has been obtained on the basis of the kernel that has been situated in the head center in the frame #10680. The second histogram has been constructed using the kernel situated in the head center in the frame #10822. Figure 6 demonstrates selected frames from the sequences which were used to construct the histograms. We can observe that despite two different realizations of an action the histograms look quite similar.

The system has also been verified on our own video data with PETS-like scenario. A high recognition ratio depending mainly on number of actions to be recognized, complication degree of actions and the way of realization of particular actions has been obtained in several dozen minutes videos. Two people performed actions such as: entering the scene and taking seat, leaving the seat, keeping seat, standing up, sitting down, walking from left to right, drawing on the board. The system achieves average recognition rate up to 90% and the frame-rate is 11-13 Hz.

8 Conclusion

We have presented an action recognition system. By employing shape, color, as well as elliptical shape features the utilized particle filter can track a head in a sequence of images and generate the trajectories of the head. The algorithm is robust to uncertainty in color representation mainly due to the fuzzy histogram based representation of the tracked head. To demonstrate the effectiveness of our approach, we have conducted several experiments using PETS-ICVS data set. One of the future research directions of the presented approach is to extend the drawing tool about a possibility of specification kernel-based zones as well as a possibility of a simulation and visualization of predefined actions.

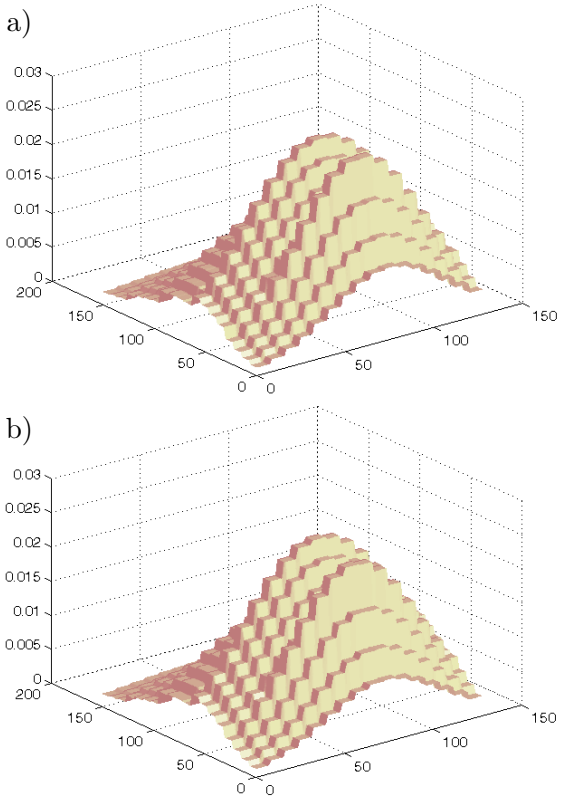

Figure 5: The kernel histograms of head positions. Scenario A, Cam 1 (a). Scenario C, Cam 1 (b).

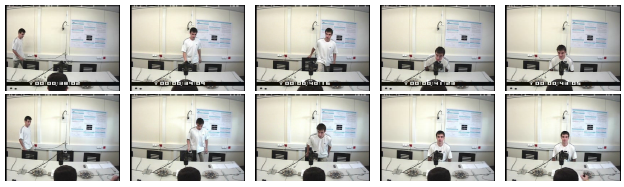

Figure 6: Selected frames from Scenario A and C. Frames #10552, #10584, #10616, #10648 and #10680 (top). Frames #10694, #10726, #10758, #10790 and #10822 (bottom).

References

[1] J. K. Aggarwal, and Q. Cai, Human motion analysis: A review, Computer Vision and Image Understanding, vol. 73, 1999, pp. 428-440.

[2] S. Birchfield, Elliptical head tracking using intensity gradients and color histograms, IEEE Conf. on Computer Vision and Pattern Recognition, Santa Barbara, 1998, pp. 232-237.

[3] I. Bloch, On fuzzy distances and their use in image processing under imprecision, Pattern Recognition, 32, 1999, pp. 1873-1895.

[4] R. L. Cannon, J. V. Dave, and J. C. Bezdek, Efficient implementation of the fuzzy *c*-means clustering algorithms, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 8, no. 1, 1986, pp. 248-256.

[5] S. Chen, and P. Gopalakrishnan, Speaker, environment and channel change detection and clustering via the Bayesian Information Criterion, Proc. Broadcast News Trans. and Understanding Workshop, 1998, pp. 127-132.

[6] J. W. Davis, and A. F. Bobick, The representation and recognition of human movement using temporal templates, Computer Vision and Pattern Recognition, 1997, pp. 928-935.

[7] A. Doucet, S. Godsill, and Ch. Andrieu, On sequential Monte Carlo sampling methods for bayesian filtering, Statistics and Computing, vol. 10, 2000, pp. 197-208.

[8] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using Mean Shift, In Proc. Int. Conf. on Computer Vision and Pattern Recognition, 2000, pp. 142-149.

[9] A. Elgammal, D. Harwood, and L. Davis, Non-parametric model for background subtraction, European Conf. on Computer Vision, vol. 2, 2000, pp. 751-767.

[10] J. Han and K. K. Ma, Fuzzy color histogram and its use in color image retrieval, IEEE Trans. on Image Processing, vol. 11, no. 8, 2002, pp. 944-952.

[11] M. Isard, and A. Blake, CONDENSATION - conditional density propagation for visual tracking, Int. Journal of Computer Vision, vol. 29, 1998, pp. 5-28.

[12] Y. A. Ivanov, A. F. Bobick, Recognition of visual activities and interactions by stochastic parsing, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, 2000, pp. 852-872.

[13] C. V. Jawahar, and A. K. Ray, Fuzzy statistics of digital images, IEEE Signal Processing Letters, vol. 3, no. 8., 1996, pp. 225-227.

[14] B. Kwolek, Stereovision-based head tracking using color and ellipse fitting in a particle filter, 8th European Conf. on Comp. Vision, LNCS, 3024, 2004, pp. 192-204.

[15] A. Madabhushi, and J. K. Aggarwal, Using head movement to recognize human activity, In Proc. of 15th Int. Conf. on Pattern Recognition, 2000, pp. 698-701.

[16] S. Mitra, S. K. Pal, and P. Mitra, Data mining in soft computing framework: a survey, IEEE Trans. on Neural Networks, vol. 13, no. 1, 2002, pp. 3-14.

[17] N. M. Oliver, B. Rosario, and A. P. Pentland, A Bayesian Computer Vision System for modeling human interactions, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, 2000, pp. 831-843.

[18] K. N. Plataniotis, and A. N. Venetsanopoulos, Color image processing and applications, Springer, 2000.

[19] P. Perez, C. Hue, J. Vermaak, and M. Gangnet, Color-based probabilistic tracking, European Conf. on Computer Vision, 2002, pp. 661-675.

[20] L. R. Rabiner, A tutorial on Hidden Markov Models and selected applications in speech recognition, Proc. of IEEE, vol. 77, no. 2, 1989, pp. 257-285.

[21] M. J. Swain, and D. H. Ballard, Color Indexing, Journal of Computer Vision, vol. 7, no. 1, 1991, pp. 11-32.

[22] J. Triesch, and Ch. von der Malsburg, Democratic integration: Self-organized integration of adaptive cues, Neural Computation, vol. 13, 2001, pp. 2049-2074.

[23] A. Tritschler, and R. Gopinath, Improved speaker segmentation and segments clustering using the Bayesian Information Criterion, In Proc EUROSPEECH, vol. 2, 1999, pp. 679-682.

[24] J. Vermaak, P. Perez, M. Gangnet, and A. Blake, Towards improved observation models for visual tracking: Adaptive adaptation, In Proc. European Conf. on Computer Vision, 2002, pp. 645-660.

[25] L. Zadech, Fuzzy sets, Information and Control, vol. 8, 1965, pp. 338-353.

[26] T. Yacoob, and M. J. Black, Parameterized modeling and recognition of activities, Int. Conf. on Computer Vision, 1998, pp. 232-247.