Paper

Face Tracking and Recognition with the Use of Particle-Filtered Local Features

Łukasz A. Stasiak[a] and Andrzej Pacut[b,c]

[a] *Fraunhofer IPK, Berlin, Germany*
[b] *Biometric Laboratories, Research and Academic Computer Network NASK, Warsaw, Poland*
[c] *Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, Poland*

Abstract—A consistent particle filtering-based framework for the purpose of parallel face tracking and recognition from video sequences is proposed. A novel approach to defining randomized, particle filtering-driven local face features for the purpose of recognition is proposed. The potential of cumulating classification decisions based on the proposed feature set definition is evaluated. By applying cumulation mechanisms to the classification results determined from single frames and with the use of particle-filtered features, good recognition rates are obtained at the minimal computational cost. The proposed framework can operate in real-time on a typical modern PC. Additionally, the application of cumulation mechanisms makes the framework resistant to brief visual distortions, such as occlusions, head rotations or face expressions. A high performance is also obtained on low resolution images (video frames). Since the framework is based on the particle filtering principle, it is easily tunable to various application requirements (security level, hardware constraints).

Keywords—biometrics, face recognition, particle filtering, video analysis.

1. Introduction

1.1. Problem Statement

We consider an identification scenario with the use of a video input signal. It is assumed that individuals entering the controlled zone cannot be effectively tracked over the entire stay-in-the-zone period, e.g., due to the large number of people walking along the main routes or due to the complicated topography of the zone. However, multiple identity recognitions with the use of local cameras installed in various locations around the controlled zone are possible. Cameras are installed in a way that enables frontal capture of subjects' faces, i.e., at the average height of a human and in specific places where frontal face images can be captured. Such *places* could be near paintings in galleries, supermarket shelves, shop windows, advertising posters, mirrors, elevator exits, escalators, at the ends of narrow corridors, etc. Based on multiple identifications, a rough track of an individual's (sequence of visited places) can be retrieved or an alarm can be raised when the selected individual enters a prohibited area in the controlled zone. It is assumed that the controlled zone is relatively small, so that the number of individuals to identify simultaneously is limited. Additionally, we assume that individuals who enter the controlled zone were previously enrolled to the system or are enrolled on entry. Consequently, a closed-set identification scenario is considered.

The presented usage scenario may be primarily regarded as *tracking by identifying* and is similar to the usage scenario of the Face Cataloger from IBM [1], [2]. Both solutions are used to answer the question *who is where?* within the controlled zone. Information gathered from tracking by identifying can be useful for warehouse, museum or gallery management, since it permits assessment of the attractiveness of the presented items. The application can also be effectively used to control higher security regions within the controlled zone, particularly, when two groups of subjects are considered. such as, e.g., employees and visitors. Unlike in the IBM's Face Cataloger scenario, we assume utilization of video-specific information not only for the purpose of tracking but also for the purpose of identity recognition. Additionally, the Face Cataloger utilizes a badge identification system for the purpose of subject identification at the entrance. We do not utilize any external systems and we assume low computational requirements for the proposed framework.

1.2. Particle Filtering

Particle filtering is used as a basis for the proposed framework and therefore it is shortly presented here. By definition, particle filters are sequential Monte Carlo methods based upon point mass representations (*particles*) of probability densities. Such representations can be applied to any state space model and generalize the traditional Kalman filtering methods [3]. The key idea of the Monte Carlo methods is to approximate a difficult analytical problem by a much simpler problem represented by a statistical sample [4]. The stochastic nature of the Monte Carlo simulation in computer environment is achieved by the use of pseudo-random number generators. The Monte Carlo simulation is considered to be one of the most influential and landmark algorithms of the 20th century [5].

An implementation of the particle filtering principle – particularly well known in the computer vision research area – is the Condensation algorithm of Isard and Blake [6]. The Condensation is also utilized within our proposed frame-

work. For the purpose of tracking, it requires at least two models to be defined, namely the object model (usually including object's dynamics model) and the observation model. Selection of the models is essential for performance of the whole solution.

Probably the most practically useful property of the particle filters is that they do not require any functional assumptions (linearity, Gaussianity, unimodality) about the densities. Initial state density $p(\vec{x}_0)$ must however be given, i.e., some initialization (e.g., initial face detection) must be done. Common drawback of the particle filtering techniques is a *degeneracy problem*, which consists in concentration of most of the weight on a relatively small subset of particles [3], [7], [8]. Full discussion of the particle filtering and degeneracy problem can be found in the cited literature.

Within the proposed framework, the particle filtering principle is utilized for the purpose of face tracking with the use of local face features (called primary face features), defined as small face patches. Color distribution within these features is analyzed for the purpose of tracking. For the purpose of recognition, the frequency analysis of the features is run. As a result, two types of secondary face features are obtained from the primary face features, namely color distribution- and frequency analysis-based secondary features. Definitions of the secondary face features as well as comments on selecting the primary features are given in the next section.

2. Local Face Features

2.1. Primary Face Features

Typically, particle filtering-based trackers assume that particle model is very similar (or identical) to the object model. In the context of the proposed framework, it would mean that particles are face candidate locations, and thus the observation area associated with each particle is of a size of the face candidate [9], [10], [11]. The final object state vector would then be calculated as mean value of all the particles. However, processing such *big* particles is usually computationally expensive. Therefore we utilize *small* particles (of a size between 10×10 and 40×40 pixels), which refer to local face patches and are understood as primary face features. Such solution results in computational time savings per each particle at the cost of the more complicated procedure of estimation of the whole face area. Namely, face area cannot be straightforwardly determined as the mean value of all particles. Instead, the distribution of all particles in image space must be analyzed to obtain rough face area estimation. For the case of many faces in the scene, this must involve automatic clustering of the particle set. Finer face area estimations are retrieved with the use of *dust filtering* procedure, which we proposed previously [12]. The dust filtering consists in classifying single pixels as skin or non-skin pixels only within the initial, roughly estimated face areas. The enhanced results of the single-pixel-classifications are then used to determine face areas more precisely. Details of the procedure of rough face area estimation from the particle distribution and of the dust filtering procedure can be found in [12].

Primary face features, i.e., particles are resampled accordingly to the Condensation schema. Each nth primary feature has assigned a weight π_n, which is used for the purpose of resampling. Additionally, random diffusion and deterministic drift are applied to steer the particle motion in the image space. For the purpose of determining the drift, a tracking history of the normalized face area locations ($\hat{R}(t)$, where $t < 0$) is stored. Predicting a new location of the face area $\hat{R}(0)$ is based on a simple model, namely

$$\hat{R}(0) = \hat{R}(-1) + (\hat{R}(-1) - \hat{R}(-2)) + \varepsilon(0), \tag{1}$$

where $\varepsilon(t)$ is an i.i.d. zero-mean noise.

2.2. Secondary Face Features for Tracking

For the purpose of face tracking, primary face features (particles) must be resampled. This is done with the use of color distribution features retrieved from the primary face features and compared to a universal skin color model. Skin color is a low level feature, which appears to be highly discriminative and computationally fast. It is easy to understand and robust to geometrical changes. As many research studies have shown, the skin tones of different ethnical groups differ mainly in their intensity values [13], [14], [15], [16], [17], being clustered in chrominance values. This makes it possible to use a universal skin color model to represent all skin types. Main disadvantage of color features is that cameras are not able to distinguish changes of the actual surface colors from changes caused by varying illumination. Consequently, illumination is the most influential factor, which changes color values recorded by a camera. Lighting compensation techniques have been proposed to reduce this problem [13], [18], [16].

In the proposed framework, color is the main cue used for the purpose of tracking by particle filtering and for the purpose of quick face normalization by means of the previously proposed dust filtering method [12]. Due to utilization of color features, real time processing can be achieved with the use of a typical modern PC. We represent color distributions of the local face features as 64×64 bin hue-saturation (HS) histograms of HSV colorspace. The V-channel (value/intensity) is ignored. We compare the HS histograms of local patches to the reference skin color model with the use of the Bhattacharyya distance $d_n = d_{Bhatt}[p_n, q]$, where p_n is the HS histogram determined from the nth particle and q is the reference color histogram. The Bhattacharyya distance is defined with the use of the Bhattacharyya coefficient $\rho[p,q]$, which is a similarity measure between two color distributions $p(u)$ and $q(u)$, namely

$$\rho[p,q] = \int \sqrt{p(u)d(u)}du. \tag{2}$$

In the context of discrete densities represented by histograms $p = \{p^{(u)}\}_{u=1...64\times64}$ and $q = \{q^{(u)}\}_{u=1...64\times64}$, the Bhattacharyya coefficient is defined as

$$\rho[p,q] = \sum_{u=1}^{64\times64} \sqrt{p^{(u)}d^{(u)}}. \tag{3}$$

For two identical normalized histograms we obtain $\rho = 1$, indicating the perfect match. The Bhattacharyya distance $d_{Bhatt}[p,q]$ is then defined as

$$d_{Bhatt}[p,q] = 1 - \rho[p,q]. \tag{4}$$

Particles (primary features) are then re-weighted accordingly to the Condensation schema: the new weight π_n of nth particle is calculated as

$$\pi_n = \exp(-\lambda\, d_n^2). \tag{5}$$

We use value of $\lambda = 20$ as suggested in [9], [19].

2.3. Secondary Face Features for Recognition

Particles which are found to be located within the dust-filtered face area, are then used for retrieval of secondary features for the purpose of identity recognition. We define these secondary features as the discrete cosine transform (DCT) coefficients of the respective primary features (particles). Whereas the secondary features for the purpose of tracking where retrieved from H- and S-channels of HSV colorspace, the secondary features for the purpose of recognition are calculated with the use of V-channel. We selected the DCT coding mainly due to its ease of application, known successful applications to face recognition [20], [21], and the potential of introducing identity recognition mechanisms into the existing compression schemes, which already utilize the DCT commonly.

Having precisely estimated the face area, relative location of each primary feature within the face area can be retrieved and thus the corresponding feature in the template can be found and compared against a given feature. This means that – for the purpose of recognition – the features are valid only in combination with their relative location within face area. The combination of frequency and location properties is similar to other existing face recognition approaches, where localization data is used in combination with some transformed local features, e.g., elastic bunch graph matching (EBGM) [22] or active shape models (ASM) [23]. However, in the previously known methods, features to be detected are precisely defined and the feature detection is the most computationally expensive part of these algorithms. In our proposed approach, we introduce primary features into the particle filtering framework. Consequently, the costly detection is skipped, *randomized* feature locations are utilized and feature sets used for the purpose of recognition differ from video frame to video frame. Such feature sets can be easily processed in real time, still providing good image exploration. Furthermore, such definition makes it possible to employ low resolution face images in which accurate detection of facial landmarks is hardly achievable [24].

The main drawback of the proposed feature set definition is that the set of stored template features should be extensive. Since any feature locations (as a result of particle filtering) can be achieved within the actually processed image, features for all possible locations within the templates should be pre-calculated. They might also be calculated on-demand, but this would lead to the high extension of the processing time. Furthermore, the face areas being processed should be well aligned with the template images, so that the actually corresponding features in the image and in the template can be compared (precise alignment or feature detection is an issue for all face recognition methods). For the purpose of testing the proposed face feature set definition approach, we utilized fast face normalization with the use of the dust filtering. Additionally, sizes of faces in test sequences were compliant with sizes of template face images. As presented in next sections, this provided good recognition rates at a low computational cost. Application of more precise normalization procedures is expected to further improve the recognition quality (at a higher computational cost).

3. Recognition from Video

A video sequence provides more information in comparison to a still face image. This information is distributed over video frames, which have some relation to the *real time* that passes during the video recording. Such distributed information can be cumulated in order to provide a stronger decision than a single-frame (or still image) based decision. The sequential hypothesis testing paradigm may be applied for the purpose of identity recognition over a sequence. The initial *weak* classifications become *stronger* when new video frames become available. An input to the recognition module are dust-filtered face areas from the tracking module. The tracking module provides consistency of the track, i.e., it assures that consecutive faces passed to the recognition module are correctly labeled as belonging to a given individual (though the individual's identity remains *unknown* to the tracking module).

As described above, secondary features from each frame are retrieved and compared to the respective secondary features of the stored templates. For the purpose of comparison, the DCT secondary features are zig-zag reorganized [25] to form feature vectors. The feature vectors can then be compared directly with the use of various distance metrics. We evaluated L_1, L_2 and L_∞ distances and L_2 resulted in the best performance. Therefore, it is used for the purpose of performance evaluation further in this article. The distance between DCT feature vectors is expressed as

$$d_{L_2}(\vec{x},\vec{y}) \quad = \quad \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}, \tag{6}$$

where $\vec{x} = [x_1,\ldots,x_n]^T$ and $\vec{y} = [y_1,\ldots,y_n]^T$ are DCT-transformed feature vectors to be compared, and $n+1$ is

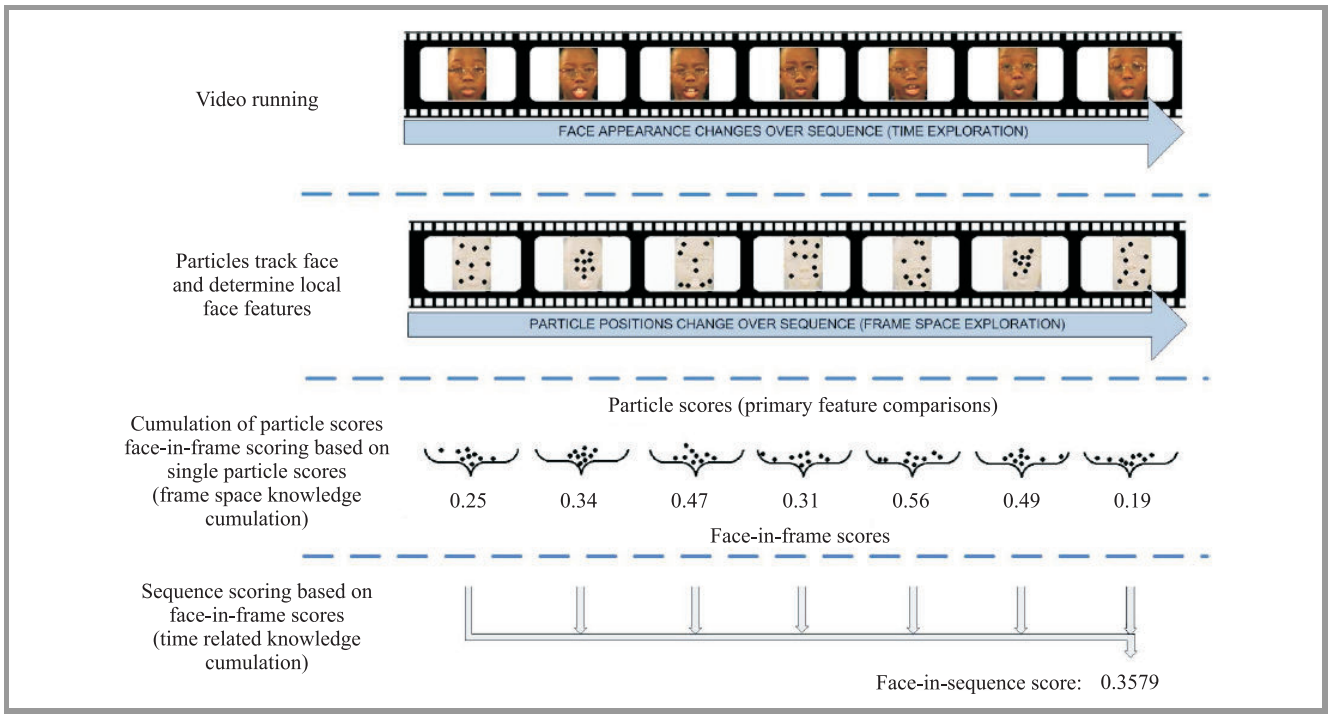

Fig. 1. Face image space and time exploration achieved by applying particle filtering to a talking head video sequence. Randomized positions of particles in each frame provide image space exploration and determine a set of face features used for the purpose of recognition. Single particle-related distances from each frame are cumulated to give a face-in-frame distance. Face-in-frame distances are cumulated to obtain a face-in-sequence distance. Sample video frames used for the presented processing were taken from [26].

the length of the feature vector (including 0 indexed coefficient). The DCT coefficient indexed 0 represents the average of the image patch and therefore is ignored for the purpose of recognition.

3.1. Cumulation of Classification Results

A result of comparing the particle-related local features is a set of distance measures between those features (retrieved from a given face image) and corresponding features of a template. These measures can be cumulated in order to provide a distance between the whole face area and the template, later referred to as the *face-in-frame distance*. The face-in-frame distance is simply calculated as an average of particle-related distance measures taken for only those particles which fall within the normalized face area, namely

$$D(F,T) = \text{avg}\left(\text{all } d_{L_2}(\vec{x},\vec{x_T}) : x \in F\right), \tag{7}$$

where $\vec{x_T}$ is a DCT feature vector in template T corresponding to DCT feature vector $\vec{x}$ retrieved from a given face image F, and x is a particle (primary feature) from which $\vec{x}$ is determined.

Face-in-frame distances are then cumulated for the purpose of comparing the whole sequences of faces against given templates, resulting in *face-in-sequence distances*. Consequently, it can be concluded that *distance cumulation* is applied at two different levels:

- Space level: cumulation of distance values (scores) of particle-related local features (distributed over a face image) in order to obtain a single face-in-frame distance.
- Time level: cumulation of face-in-frame distance values (distributed over a video sequence) in order to obtain a face-in-sequence distance.

Comparisons of local features can also be understood as weak local classifications, which are then cumulated to provide stronger frame-related classifications (face-in-frame distances). Cumulation of frame-related distances provides yet stronger classification of video sequences (face-in-sequence distances). Face image space exploration is a result of the integrated framework by which the probabilistic nature of the particle filtering-based tracking is passed to the recognition task. Consequently, more face image space can be explored and tested for the purpose of recognition without significantly increasing the processing burden. Two levels of distance cumulation within the proposed framework are depicted in Fig. 1. The face-in-sequence distances are then utilized for the purpose of recognition, which is done by building a ranking of identities. Since we consider a closed set scenario, all stored templates are compared (scanned) against the actually processed face image and the ranking is determined.

Integration of the tracking and recognition within one framework brings additional advantage, namely identity cue can be used for the purpose of tracking corrections. In case of processing multiple faces in the scene, some face-to-track assignment conflicts occur. A feedback from the

recognition module can additionally support conflict resolution mechanisms and thus improve tracking accuracy. We described this idea in [27].

3.2. Multi-Image Template

Quality of a template has a great influence on the overall recognition performance. The template quality can be improved by using more than one image for template creation. The simple template extension technique, which directly uses several images to create a multi-image template, can be used: the query face image F is compared to each image of the multi-image template T^* and the best match between the query image and template image is selected as the actual face to multi-image template distance D^*, namely, for K-image multi-image template

$$D^*(F,T^*) = \min_{1\le k\le K} D(F,T_k^*), \tag{8}$$

where T_k^* is kth image of the multi-image template and D is the face-in-frame distance calculated between the given face and a single template image. Performance improvement achieved by extending the template representation and with the use of the simple comparison procedure is analyzed in the following sections.

4. Definition of the Framework and Its Parameters

4.1. Testing Environment

For the purpose of testing the performance of the proposed framework, we used 55 video sequences of the Open Video Project (OVP) [26]. The downloaded sequences are *talking heads* videos of different individuals. The length of sequences varies from 851 to 8265 frames. Talking head videos of 35 different individuals have been extracted, resulting in 100 to 1187 talking head frames per individual (340 talking head frames per individual on average). Talking head sequences contain frontal and almost-to-frontal shots (less than 30 degrees profile) without any extra constraints on the individual's head motion or facial expression. Changes in captured head size due to camera zoom or head motions are present. A few sample frames from the test video sequences are presented in Fig. 2.

Fig. 2. Variations in test video sequences from Open Video Project [26]. Test videos are frontal and almost-to-frontal talking head sequences without any additional constraints on the individual's head motion and facial expression. Variations in head size, video quality and background type are noticeable.

It is to notice that – since we consider a sequential recognition from video sequences – the beneath reported cumulative match characteristics (CMC) are *cumulated* not only over identities but also over sequence time. For example, the 80% 1-rank identification rate means that the actual subject identity was returned in the first position in the ranking for 80% of the time (video frames) in all test sequences. Various aspects of the proposed framework have been evaluated and the results are presented beneath.

4.2. Optimal Feature Vectors

Selection of DCT coefficients for the purpose of building feature vectors for identity recognition can influence the overall recognition performance. The DCT coefficients selection is related to the question of how much identity-specific information is carried by various signal frequencies. Ekenel and Stiefelhagen [28] showed that selection of the number of coefficients influences performance and that extending this number over a certain limit does not significantly improve performance. Sanderson et al. [29] showed that increasing the dimensionality from 15 to 21 provides only a small recognition improvement, while it significantly increases the computational requirements.

In order to find an optimal set of DCT coefficients we run several tests for the closed-set scenario on the full test database, but with the use of different feature vector definitions. The testing was done with the use of the whole proposed framework. Although all framework parameters will only be introduced hereinafter, we think it is reasonable to present the results regarding selection of the DCT

Table 1
Influence of the number of DCT coefficients on the identification rates obtained for full database testing in the closed-set scenario

Indexes of selected DCT coefficients	Identification rate [%]		
	1-rank	5-rank	10-rank
1-5	66.64	79.49	85.66
1-10	65.15	79.15	86.21
1-15	64.16	78.92	85.70
1-20	62.16	77.50	84.74
1-25	60.48	76.54	83.89
6-15	41.24	60.27	74.74
16-25	22.70	44.58	59.92

coefficients here. For the purpose of this testing fixed particle positions were used, so that the results of different test runs (for different feature vector definitions) are comparable. The identification rates obtained with the use of differently defined feature vectors are gathered in Table 1 and depicted in Fig. 3.

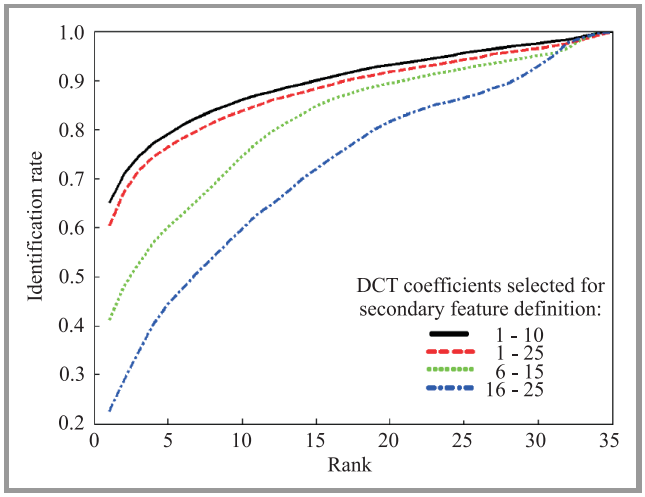

Fig. 3. Influence of the feature vector definition on CMCs obtained from full database testing in the closed set scenario.

The obtained results show that lower frequencies (low coefficient indexes) contain most of the identity-related information. Extending the feature vector by higher frequencies does not improve the performance significantly and excluding lower frequencies drastically reduces performance quality. It can also be observed that differences between the cases of 1-5, 1-10, 1-15, 1-20 and 1-25 are minor. However, in the literature [30], [28], [29], it is rare that as few as five coefficients are suggested. We finally selected DCT coefficients 1-10 to be used as the local feature representation in our framework. Such a definition results in good recognition performance and keeps the representation compact and is used for the evaluation presented in the following sections. It is to comment that before calculating the DCT coefficients, contrast of the whole face area (in a given frame and in a template) is enhanced by the histogram equalization technique, which improves system performance [28]. No other illumination compensation techniques are applied.

4.3. Face-in-Sequence Scoring and Classification

Having obtained face-in-frame distances against a set of templates, a cumulated distance for the video sequence, i.e., face-in-sequence distance, can be obtained. Cumulation of the distances can be done in the following ways:

- Fixed lag cumulation. Face-in-sequence distance is based on the distances of n previous frames (cumulation lag = n). Results are available at any ith frame (time) of the sequence, such as $i > n$.
- Fixed point (growing lag) cumulation. Face-in-sequence distance is based on all previous frames. Results are available at any frame of the sequence.
- Adaptive lag cumulation. Face-in-sequence distance is based on the varying number of previous frames. Results may be available at any frame of the sequence (but with different *strength*) or when a given minimal number of frames is available.

In all cases *previous frames* must be understood as face areas retrieved from previous frames and with reference to a given track.

In order to achieve good classification when large lag values are used (a high number of previous frames is considered), it must be ensured that tracks are consistent, i.e., the subject-track pairs are not swapped during tracking. Otherwise, classification of a sequence (track) containing face images of various subjects will be dominated by the prevailing subject. In consequence, application of the fixed point scoring is not appropriate for high security scenarios and should rather be applied to other non-security scenarios, e.g., for the purpose of video summarization [31]. For security applications, utilization of the fixed or adaptive lag is more appropriate. The lag value does not only influence classification strength, but it also defines response delay (e.g., updating identity classification result or raising an alarm), when the subject identity within a track changes (which, first of all, may be a result of tracking error). Response delay can also be understood as a resistance to brief misclassifications: the higher the lag value, the more the duration of the misclassification (e.g., caused by occlusion) will not affect classification result. The trade-off between a quick response to identity change and the resistance to misclassifications is actually the problem of tuning a biometric system to achieve optimal false acceptance and false rejection rates (FAR and FRR). An optimal solution does not seem to exist in general and should be found with respect to application specific requirements, such as security level, environmental conditions, input video quality, usability requirements, hardware requirements (e.g., memory requirements for storing previous frame distances). It may be concluded that optimally the lag value should change within some predefined range $[\mathrm{lag}_{\min}, \mathrm{lag}_{\max}]$. The value of $\mathrm{lag}_{\min}$ should be derived from the required minimal classification strength and misclassification-resistance, whereas $\mathrm{lag}_{\max}$ should be derived from maximal acceptable response delay.

Level of the face-in-frame distances is dependent on the input frame conditions, such as e.g. head rotation or frame quality. It means, that though the ranking of identities can be preserved between the frames, the absolute level of distance values can vary strongly and influence the cumulated distance. Therefore, for building the cumulated rankings, a distance value normalization is required. For this purpose we utilize min-max normalization of face-in-frame distances, namely

$$D'_{FT} = \frac{D_{FT} - D_{F_{\min}}}{D_{F_{\max}} - D_{F_{\min}}}, \tag{9}$$

where: $D_{FT} = D(F,T)$ is the calculated (unnormalized) distance between frame F and template T, D'_{FT} is the normalized frame-to-template distance, $D_{F_{\min}}$ and $D_{F_{\max}}$ are respectively minimal and maximal distances between the given frame F and any template from the template set. As a result of normalization the values of D'_{FT} within the range of $[0,1]$ are obtained.

4.4. Occurrence of Classification Errors

During evaluation we have observed that erroneous or *weak* classifications are usually a result of distortions in video sequence. In such cases most of the templates seem to be *almost equally* distant to the given frame. In other words, the given frame is not particularly similar to any given template. On the other hand, if the recognition is strong, there are usually only a few *good* matches being clearly separated from others. This effect is illustrated in Fig. 4.

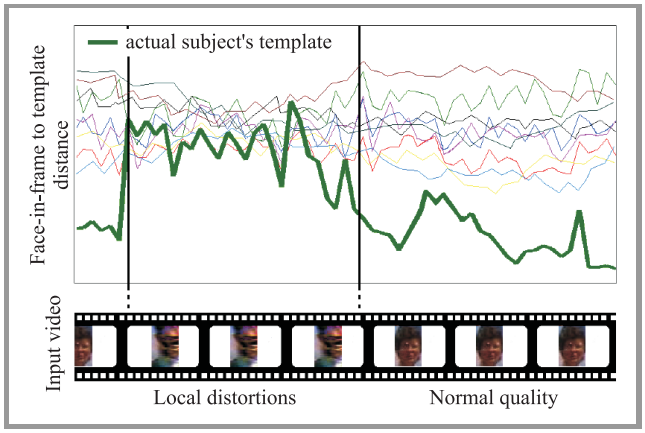

Fig. 4. Local distortions in video cause face-in-frame distance values to *gather* around an average value. In good quality frames, best-matches are clearly separated from other matches. Here distortions are caused by digital storage medium errors.

Based on this observation it may be concluded that strong classifications are possible when the subject's face in the video can be seen well. Since *bad* matches are similarly bad for all the templates, distance value cumulation should enable the brief erroneous classifications to be overcome when a longer period of time is considered.

5. Distance Value Cumulation Mechanisms

5.1. Utilizing Video Sequentiality

By video sequentiality we mean the high dependency of a video frame on previous frames. It may be informally said that almost every video frame is very similar to the preceding frame. In the identity recognition context the sequentiality may be utilized to overcome brief misclassifications, since an identity recognized in a given frame is very likely to have also appeared in previous frames. This property can be utilized by applying a cumulation mechanism, i.e., ranking the identities of each video frame with respect to previous frames. A simple approach is to use the sum of face-in-frame distances on a lag of k previous frames as a cumulated distance. Namely, the cumulated distance D_{cum} for the jth frame F_j of the sequence against a given template T on a lag of k previous frames is defined as:

$$D_{cum}(F_j,T) = \frac{1}{k}\sum_{i=0}^{k-1} D(F_{j-i},T). \tag{10}$$

Implementing the cumulation mechanism results in a higher recognition rate (as calculated per every video frame). Similarly, extending template representation from one image to a three image multi-image template increases recognition rates. The observed performance improvement is summarized in Table 2 and depicted in Fig. 5. It is observed that increasing the template quality improves the performance more than introducing cumulation mechanism only.

Table 2
Identity recognition improvement obtained as a result of introducing cumulation mechanism (with lag $L = 10$) and extending template representation: 1-rank identification rates presented

Solution	1-rank ident. rate [%]
One-image templates, no cumulation	34
One-image templates with cumulation	41
Three-image templates, no cumulation	50
Three-image templates with cumulation	61

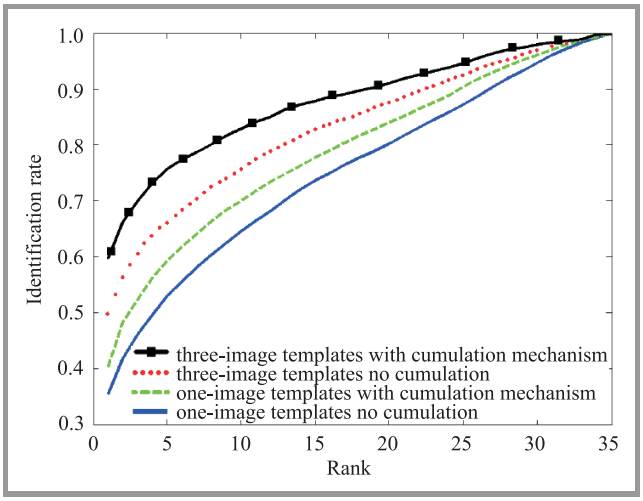

Fig. 5. Identity recognition improvement obtained as a result of introducing cumulation mechanism (with lag $L = 10$) and extending template representation: CMCs depicted.

5.2. Cumulation Schemes

Fusion of face-in-frame distances can involve a simple sum rule or can be combined with extra distance value transformation. Let us denote D as the original distance determined

by scoring a given face-in-frame and D' as the distance after transformation. We propose and evaluate the following transformation methods:

– linear mapping $D' = D$, (11)

– square root transformation $D' = \sqrt{D}$, (12)

– quadratic transformation $D' = D^2$. (13)

Linear mapping is used by a basic sum rule approach: it corresponds to a simple summation of all frame distances over the sequence. Square root and quadratic transformations are meant to emphasize Eq. (12) or de-emphasize Eq. (13) differences between *similarly good* matches. As described above, all face-in-frame distances are min-max normalized before applying transformations of Eqs. (11), (12), (13).

CMCs for different fusion approaches with the cumulation lag of $L = 10$ and three-image template representations are depicted in Fig. 6. The best performer, namely the square root fusion approach, achieved **1-rank** identification rate of **61%**, **5-rank** rate of **77%** and **10-rank** rate of **85%**. The simple sum rule (linear mapping) performed almost equally well.

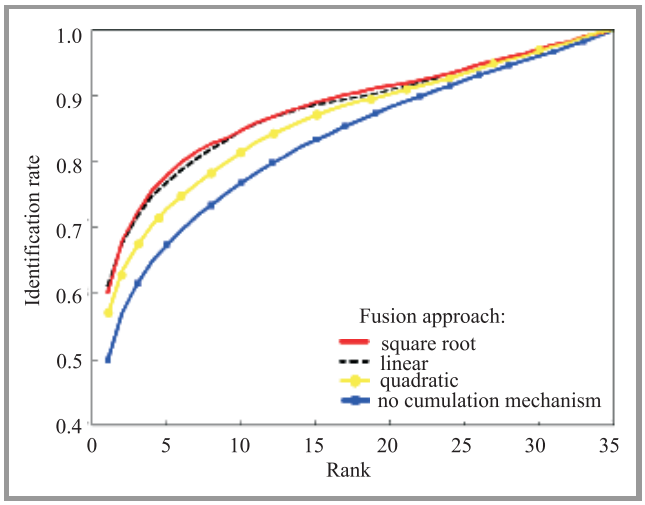

Fig. 6. CMCs obtained with the use of different fusion schemes, cumulation lag of $L = 10$ frames and three-image template representations. Application of cumulation mechanisms significantly improves performance in comparison to the non-cumulation approach.

It is again observed that application of any cumulation mechanism improves the recognition performance significantly in comparison to single frame based identification. It is also concluded that the simplest summation rule may be optimal solution, since differences in performance quality between simple rule and square root fusion are minor.

5.3. Definition of the Cumulation Lag and the Influence of Input Video Frequency

During evaluation it was discovered that defining the cumulation lag by the number of frames was confusing – it is rather the time period which should be defined as a cumulation lag. Time-based definition of the cumulation lag remains independent of video frequency, unlike definition by number of frames. Time-based definition describes the admissible range of appearance changes better than does the number of frames. We observed that given a time defined cumulation lag, changes in recognition performance caused by different input video frequencies are minor and can be disregarded. In other words, the most influential factor on identification rates is the period of distortions in video in relation to the cumulation lag (i.e., how long can distortions maximally last), and this property is easier to describe by defining the cumulation lag in time units.

5.4. Optimal Lag Value Selection

Selection of the lag value L determines the recognition performance of the system and its response-delay to the *identity change* of the observed individual. Choice of an optimal lag value is application-specific but a general policy can be defined:

- For high security the lag value should be low to provide a quick response.
- For high user-friendliness the lag value should be higher to minimize the number of false rejections.

To evaluate influence of the lag L on the overall performance, we evaluated the proposed framework with the use of various cumulation lag values. Lag $L = 1$ is equivalent to the case with no cumulation mechanism. For this configuration the 1-rank identification rate of 50% was obtained, which confirms the *weak* nature of the used frame classifier. For a cumulation lag equal to 8 s (200 frames) **1-rank** identification rate of **81%**, **5-rank** identification rate of **90%** and **10-rank** identification rate of **93%** were obtained. CMCs for various cumulation lag values with the use of the square root fusion approach Eq. (12) are depicted in Fig. 7.

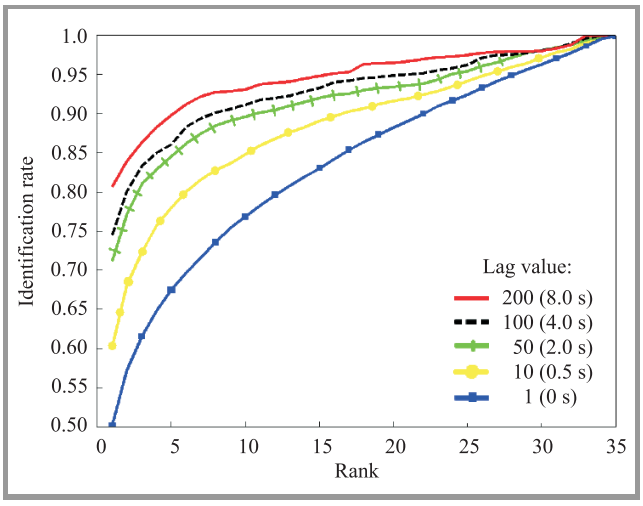

Fig. 7. CMCs for the square root fusion approach and various cumulation lag values L. Increasing the L value improves recognition performance. Graphs for 25 fps sequences.

The obtained results confirm expectation that extending the cumulation lag improves the recognition performance of the framework. It is also observed that cumulating only a few previous frame distances improves performance significantly. The more frame distances are already cumulated, the less influence on performance is observed by adding new distances. It is however to remember, that no track swapping errors were considered during this test: i.e., an indexed track have never *skipped* to another individual (assumption of tracks consistency). In the target application, as mentioned previously, not only recognition rate, but also response delay must be considered when selecting the optimal lag value.

5.5. Fixed Point Approach

Increasing the lag improves recognition performance. As a result, it may be expected that the fixed point (growing lag) cumulation approach would provide higher identification rates. The drawbacks of this approach, as described previously, include the risk of high response delay when a tracking error occurs. For the purpose of testing the progress of identification rate in fixed point approach we extracted a subset of 100-frame long sub-sequences from the testing database. Rank 1 and rank 5 recognition rate changes observed over frames of the extracted video sequences are depicted in Fig. 8.

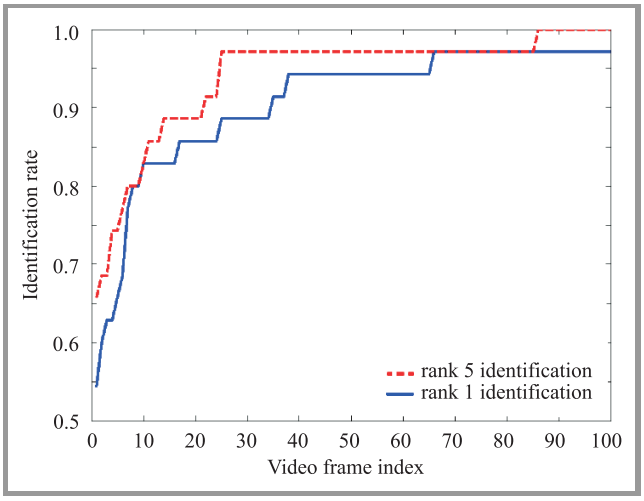

Fig. 8. Rank 1 and rank 5 identification rates over 100-frame sequences with the fixed point cumulation approach (all previous frame distances considered).

From the results it can be concluded that extending the cumulation lag does not need to be indefinite. In early processing steps, as the cumulation lag value is low, the recognition performance increases rapidly with new frames. Later on however, as many previous frames are already considered, the cumulated score becomes saturated. This leads again to the idea of utilizing an adaptive lag as the most practical approach. It should be remembered that this test was run with sub-sequences of the original testing database, therefore the reported results vary from those presented previously for the whole database.

5.6. Processing Times and Further Enhancements

The proposed framework was tested on an Intel Core 2 Duo E6750 computer, 2.67 GHz with 2.00 GB RAM. Processing times were tested in a whole framework combined with a detection module, which is not described in this paper and which initializes the tracking process. The detection, tracking and normalization functionalities were implemented in the Visual C++ environment and with the use of OpenCV 1.0 library [32]. The recognition module was implemented in Matlab 6.5. No special code optimization was applied.

Processing times were calculated for the following configuration parameters and environmental conditions: input video frame of size 320×240 pixels, particles of size 8×8 pixels, normalized face areas of size 64×64 pixels, face detection by Haar-like face detector [33], generic skin color model represented by 64×64 hue-saturation histogram, face area normalization by means of dust filtering with pixel-step equal to 4, 10-subject closed set identification scenario, three-image template representations, one person in the scene.

In the basic configuration, 335 ms per frame were needed to run all the tasks of detection, tracking and recognition with the use of 50 particles. Face detection was the most time consuming task – tracking and recognition itself needed 85 ms per frame on average. Frame preprocessing, which involved data retrieval from the video buffer and transformation from RGB to HSV colorspace, required 3 ms of the processing time.

Some process optimizations, which should further reduce computational requirements, are possible. First of all, face-in-frame to template comparisons are currently realized by linear scanning of the whole set of templates. Therefore, the recognition processing time is proportional to the number of subjects in the database. Effective indexing and sorting techniques are subject to further research with the aim of ensuring that a quick search during identification can be carried out.

Additionally, average cost of face detection can be reduced by minimizing the frequency of running the detection process. The detection can be, for example, triggered by an external event, such as door open etc. Furthermore, tracking and recognition do not need to operate on every single frame, but can wait until detection is finished. This would result in the longest processing time per frame of 250 ms occurring during the detection phase. After detection, the whole frame processing would require **85 ms** (tracking and recognition only) – this means that a speed of **11 fps** can be achieved and 5 fps is regarded as sufficient for handling *normal* head motions [34]. For the purpose of detection in a testing environment the frontal face detector and profile face detector were utilized. Profile detection involved horizontal mirroring of the whole frame. Reducing detection to frontal faces only can save 190 ms of processing time.

Consequently, simultaneous frontal face detection, tracking and recognition in every frame results in a processing speed of more than **6 fps**. A lower accuracy in detection may be accepted for many applications, in particular due to utilizing video as an input signal: in many practical cases it may be reasonable to invoke detection less accurately (i.e., detecting frontal faces only), but more frequently. Detection by tracking approach [12], [35] can also be utilized to further reduce processing times. Achieving the optimal architecture of the modules (sub-processes) is, however, a non-trivial and application-specific issue.

The advantages of using the distributed hardware architecture should be considered for the proposed framework. Due to construction of sub-processes, detection can be easily realized by other processing units than tracking and recognition. Separating tracking and recognition between different processing units is also possible. In the distributed environment each process would run independently and retrieve required data from the supporting process (e.g., tracking from detection or recognition from tracking). A further degree of parallelization could be achieved by computing the DCT for various particles on separate units.

6. Conclusions

We proposed and evaluated a consistent particle filtering-based framework for face tracking and recognition from video. Presented results proved that sequentiality of the video signal can be effectively used for the purpose of increasing identification rates. This is achieved by applying distance cumulation mechanisms. Even utilization of weak classifiers, which result in the 1-rank identification rate of 50% when no cumulation mechanism is applied, can lead to 1-rank identification rate of 81% when a cumulation lag of 8 s (200 frames) is used. The strength of the classification increases as more frame distances are collected for the purpose of classification. The classification result is available at any video frame, so it can be obtained even at early steps of the sequence processing, though with lower accuracy. The number of previous video frames used for the purpose of classification, i.e., the cumulation lag, can be adapted to the needs of a particular application.

The proposed particle filtering-based determination of local face features enables good exploration of a face space over a video sequence and results in high recognition performance, while keeping computational requirements at a modest level. Consequently, real-time processing can be achieved on an ordinary modern PC. The trade-off between quality and computational requirements can easily be optimized for the purpose of specific applications by tuning the number of particles. Additional tuning is possible by adapting the cumulation lag to given environmental conditions or application requirements.

A particle filtering-based definition of a face feature set, in combination with cumulation mechanisms, is resistant to small rotation- and expression-caused appearance changes. It also provides good recognition performance from low resolution input videos and performs well in combination with fast dust filtering-based face normalization (in low resolution videos, precise classical normalization, such as that based on eye positions, is often not possible at all).

The proposed system opens new fields for future research. One of the most promising directions is integration of our solution with speaker recognition technology. Both approaches can operate on data retrieved from talking head video sequences, provided that voice is recorded. The integration should ensure mutual support between face-based and voice-based recognition, in particular in cases when one of two signal sources becomes unclear or temporarily unavailable. Hardware focused research should also be conducted in order to examine the advantages of the parallelizing sub-processes of the proposed system, and thus decrease the overall processing time. Additionally, research on the usage of particle filtering-determined (randomized) feature sets for the purpose of recognition in other scenarios, including recognition from still images, should provide interesting conclusions for applications with limitations on processing time. In-depth examination of distance fusion schemes – both at the frame level and sequence level – could bring further performance improvements. The possibility of using other classifiers (instead of Euclidean distance-based one) to determine face-in-frame distances should be examined. Other secondary feature representations (apart from DCT), e.g., derived from training-based methods, such as the PCA, should also be evaluated within the proposed framework.

Acknowledgements

The work of the second Author has been financed by the Ministry of Science and Higher Education grant OR00 0026 07 "*A platform of secure biometrics implementations in personal verification and identification*".

References

[1] A. Hampapur, L. Brown, J. Connell, A. Ekin, N. Haas, M. Lu, H. Merkl, S. Pankanti, A. Senior, C.-F. Shu, and Y. L. Tian, "Smart video surveillance – exploring the concept of multiscale spatiotemporal tracking", *IEEE Sig.. Proces. Mag.*, vol. 22, pp. 38–51, 2005.

[2] "IBM Smart Surveillance System", 2009 [Online]. Available: http://www.research.ibm.com/peoplevision

[3] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking", *IEEE Trans. Sig. Process.*, vol. 50, no. 2, pp. 174–188, 2002.

[4] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning", *Machine Learn.*, vol. 50, no. 1-2, pp. 5–43, 2003.

[5] J. Dongarra and F. Sullivan, "The top 10 algorithms", *Comput. Sci. Eng.*, vol. 2, pp. 2–97, 2000.

[6] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking", *Int. J. Comp. Vis.*, vol. 2, pp. 5–28, 1998.

[7] P. Perez, J. Vermaak, and A. Blake, "Data fusion for visual tracking with particles", *Proc. IEEE*, vol. 92, no. 3, pp. 495–513, 2004.

[8] A. Doucet, N. de Freitas, and N. Gordon, "An introduction to sequential Monte Carlo methods" in *Sequential Monte Carlo Methods in Practice*, New York: Springer, 2001, pp. 3–14.

[9] P. Perez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking", in *Proc. 7th Eur. Conf. Comp. Vis. ECCV 2002*, Copenhagen, Denmark, 2002.

[10] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time tracking of non-rigid objects using mean shift", in *Proc. IEEE Int. Conf. Comp. Vis. Patt. Recogn. CVPR 2000*, Hilton Head Island, USA, 2000, pp. 142–149.

[11] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter", *Image and Vis. Comput.*, vol. 21, no. 1, pp. 99–110, 2003.

[12] Ł. Stasiak and A. Pacut, "Particle filtering in multilevel color context for face detection and tracking in real-time video sequences", in *Proc. 42nd Ann. IEEE Int. Carnahan Conf. Secur. Technol. ICCST 2008*, Prague, Czech Republic, 2008.

[13] R.-L. Hsu, M. Abdel-Mottaleb, and A. K. Jain, "Face detection in color images", *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 24, no. 5, pp. 696–706, 2002.

[14] S. Spors and R. Rabenstein, "A real-time face tracker for color video", in *Proc. IEEE Int. Conf. Acoust. Speech. Sig. Proces. ICASSP'01*, Salt Lake City, USA, 2001.

[15] V. Vezhnevets, V. Sazonov, and A. Andreeva, "A survey on pixel-based skin color detection techniques" in *Proc. Graphicon-2003*, Moscow, Russia, 2003.

[16] J. Yang, W. Lu, and A. Weibel, "Skin-color modeling and adaptation". Tech. Rep. CMU-CS-97-146, School of Computer Science, Carnegie Mellon University, May 1997.

[17] J. Birgitta Martinkauppi and M. Pietikainen, "Facial skin color modeling" in *Handbook of Face Recognition*, S. Z. Li and A. K. Jain, Eds. New York: Springer, 2005, pp. 113–135.

[18] J. Yang and A. Waibel, "Tracking human faces in real-time", Techn. Rep., CMU-CS-95-210, School of Computer Science, Carnegie Mellon University, Pittsburgh, November 1995.

[19] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: multitarget detection and tracking", in *Proc. 8th Eur. Conf. ECCV 2004*, Prague, Czech Republic, 2004.

[20] Z. M. Hafed and M. D. Levine, "Face recognition using the discrete cosine transform", *Int. J. Comp. Vis.*, vol. 43, no. 3, pp. 167–188, 2001.

[21] Y. Chen and Y. Zhao, "Face recognition using dct and hierarchical RBF model", in *Proc. 7th Int. Conf. IDEAL 2006*, Burgos, Spain, 2006, *Lect. Not. Comp. Sci.*, vol. 4224, pp. 355–362.

[22] L. Wiskott, J.-M. Fellous, N. Krueger, and C. von der Malsburg, "Face Recognition by Elastic Bunch Graph Matching", in *Intelligent Biometric Techniques in Fingerprint and Face Recognition*, L. C. Jain, U. Halici, I. Hayashi, S. B. Lee, and S. Tsutsi, Eds. Boca Raton: CRC Press, 1999, pp. 355–396.

[23] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models – their training and application", *Comp. Vis. Image Understand.*, vol. 61, no. 1, pp. 38–59, 1995.

[24] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: a literature survey", *ACM Comput. Sur.* , vol. 35, no. 4, pp. 399–458, 2003.

[25] C. Fogg, "Questions that should be frequently asked about MPEG, April 1996 [Online]. Available: http://bmrc.berkeley.edu/research/mpeg/faq/mpeg2-v38/faq_v38.html

[26] "The Open Video Project" [Online]. Available: http://www.open-video.org/

[27] Ł. A. Stasiak and R. Vicente-Garcia, "Identity recognition-based correction mechanism for face tracking", in *Proc. 6th Conf. AMDO*, Andratx, Spain, 2010.

[28] H. K. Ekenel and R. Stiefelhagen, "Local appearance based face recognition using discrete cosine transform", in *Proc. 13th Eur. Conf. EUSIPCO 2005*, Antalya, Turkey, 2005. *Sig. Proces.*, vol. 83, no. 5, pp. 931–940, 2003.

[29] C. Sanderson and K. K. Paliwal, "Features for robust face-based identity verification",

[30] J. Stallkamp, H. K. Ekenel, and R. Stiefelhagen, "Video-based face recognition on real-world data", in *Proc. IEEE 11th Int. Conf. ICCV 2007*, Rio de Janeiro, Brazil, 2007, pp. 1–8.

[31] Y. Li, T. Zhang, and D. Tretter, "An overview of video abstraction techniques", Tech. Rep. HP-2001-191, HP Laboratory, July 2001.

[32] "Open computer vision library" [Online]. Available: http://sourceforge.net/projects/opencvlibrary/

[33] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features", in *Proc. IEEE Int. Conf. CVPR 2001*, Kauai, USA, 2001, pp. 511–518.

[34] S. Z. Li and A. K. Jain, Eds., *Handbook of Face Recognition*. New York: Springer, 2005.

[35] Ł. Stasiak and A. Pacut, "Particle filters for multi-face detection and tracking with automatic clustering", in *Proc. IEEE Int. Worksh. Imag. Sys. Techniq.*, Cracow, Poland, 2007.

Łukasz A. Stasiak was born in 1980 in Warsaw, Poland. He received the M.Sc. (with honors) in computer science from Warsaw University of Technology in 2004. From 2003 to 2006 and in 2008 he was with Biometric Laboratories, Research and Academic Computer Network NASK, Warsaw, Poland as a researcher in the field of biometric systems, with particular interest in hand- and face-based methods. He was involved in creation of first scientific biometric database in Poland. He also participated in development of systems for iris and hand signature recognition. In 2007 and 2008 he worked in a software company in Warsaw in area of Geographic Information Systems (GIS). From 2009 he is with the department for Security Technology at the Fraunhofer-Institute for Production Systems and Design Technology IPK in Berlin, Germany.
e-mail: lukasz.stasiak@ipk.fraunhofer.de
Fraunhofer IPK
Pascalstrasse 8-9
10587 Berlin, Germany

Andrzej Pacut – for biography, see this issue, p. 18.