Probabilistic Temporal Head Pose Estimation Using a Hierarchical Graphical Model

Meltem Demirkus, Doina Precup, James J. Clark, and Tal Arbel

Centre for Intelligent Machines, McGill University, Montreal, Canada

Abstract. We present a hierarchical graphical model to probabilistically estimate head pose angles from real-world videos, that leverages the temporal pose information over video frames. The proposed model employs a number of complementary facial features, and performs feature level, probabilistic classifier level and temporal level fusion. Extensive experiments are performed to analyze the pose estimation performance for different combination of features, different levels of the proposed hierarchical model and for different face databases. Experiments show that the proposed head pose model improves on the current state-of-the-art for the unconstrained McGillFaces [10] and the constrained CMU Multi-PIE [14] databases, increasing the pose classification accuracy compared to the current top performing method by 19.38% and 19.89%, respectively.

Keywords: Face, hierarchical, probabilistic, video, graphical, temporal, head pose.

1 Introduction

Video cameras are ubiquitous in today's world, from street and area surveillance to intelligent digital signs and kiosks. The imagery provided by these cameras is unconstrained and capture video streams of people in many different poses and under a wide variety of lighting conditions. Robustly estimating head pose from such video is an increasingly important and necessary task. In the context of real-world scenarios, face recognition/verification, facial attribute classification and human computer interaction all generally benefit from using head pose estimates as prior information in order to boost their performance [7, 12, 16, 19, 22, 23, 41].

There is a wide literature on head pose estimation, [2–4, 6, 11, 13, 17, 18, 20, 25, 27, 28, 30, 32, 34, 35, 39, 42, 43]. The general categories of methods described in this literature include [26]: *Appearance template methods* use image-based comparison techniques to match a test image to a set of training images with corresponding pose labels. *Manifold, subspace embedding methods* project an image onto the head pose manifold using linear and nonlinear subspace techniques. When such techniques are used for video frames, they implicitly model a given video sequence temporally by mapping similar frames onto nearby locations in the manifold. *Geometric methods* use the location of facial landmarks to determine the head pose from their relative configuration. Lastly, *tracking methods* aim to estimate the global movement of a head by using the relative movement

D. Fleet et al. (Eds.): ECCV 2014, Part I, LNCS 8689, pp. 328–344, 2014.

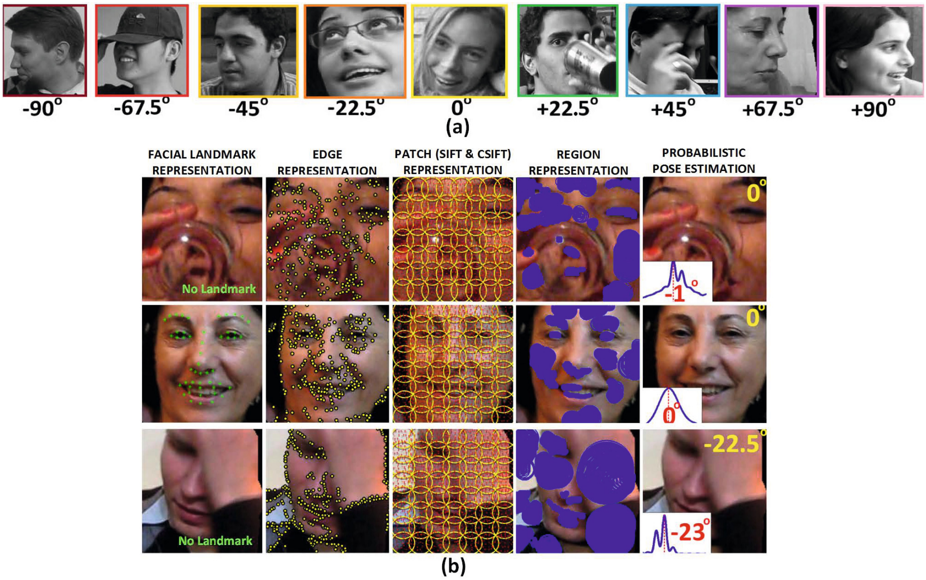

Fig. 1. Examples of tracked faces from the McGillFaces Database [10] depicting the head pose ground truth angles, extracted features and estimated pose information: (a) Head pose (yaw angle) ground truth labels of example faces, (b) Different facial representations, namely facial landmark [40], facial region (BPLR), patch (SIFT and CSIFT) and edge (GB), employed by the proposed model and the estimated pose information. The pose ground truth label obtained via [10] is shown on the top right. The yaw distribution calculated by our approach is shown on the top bottom and the corresponding MAP estimate is shown in red on the pose distribution.

between consecutive video frames. The highest accuracies published in the head pose literature are presented by the manifold learning methods, e.g.[4]. Most of these methods, however, are not designed to operate in unconstrained environments. A common assumption is that the entire set of facial features typical for frontal poses is always visible. Facial features are often manually labeled in the testing data, rather than extracted automatically. Furthermore, most approaches are trained and tested on images which do not exhibit wide variation in appearance. The testing databases mostly contain images with solid or constant background, limited facial expression, no random illumination, and limited or no facial occlusion (e.g. Multi-PIE [14]). Finally, current face tracking methods require a known initial head pose, and usually must be reinitialized whenever the tracking fails. All of these issues contribute to poor performance when applied to real-world videos.

Estimation of head pose from uncontrolled environments has recently been receiving more attention [2, 11, 12, 28, 35, 40, 43]. Orozco et al. [28] and Tosato et al. [35] address the problem of head pose estimation in single, low resolution video frames of crowded scenes under poor and/or limited, e.g. indoor, lighting, where they treat the problem as a classification problem. That is, they assign a face image to one of the coarse discrete pose bins, e.g. front and back. Some approaches [11, 43], on the other hand, use relatively higher quality video frames/images and perform classification on finer pose bins whereas others defined the pose estimation problem as a continuous (fine discrete) pose angle estimation task [2, 12, 40]. In short, most of these approaches either do not leverage the temporal pose information available between consecutive video frames, or focus on only a specific set of features (e.g. facial landmark points) to represent faces. It is shown in [40] that facial landmarks can be used successfully to estimate head pose when they are reliably located. However, it is difficult to extract such features when a significant facial occlusion is present (see Fig. 1(b)) or when the pose angle is more than 45^o, leading to occlusion of facial landmark regions (e.g. eyes) in the image.

This paper is concerned with the automated estimation of very fine discrete head pose (yaw angle only) in unconstrained videos. The video data is assumed to include difficult aspects such as a wide range in face scales, extreme head poses, variable and non-uniform illumination conditions, partial occlusions, motion blur, and background clutter (see Fig. 1). The probabilistic graphical model proposed in this paper (Fig. 2 and 3) is based on a hierarchy of complementary robust local invariant facial features, which leverages the dependencies between consecutive video frames in order to substantially improve head pose estimation in real world scenarios. These features have a high degree of invariance to various transformations, such as changes in scale, viewpoint, rotation and translation. They include: (i) facial landmarks, (ii) densely sampled patch-based features, (iii) regions, mainly associated with anatomical structures such as the eyes, forehead, and cheeks, and (vi) edge points, mainly arising from the eyebrows, mouth, eyes and nose (see Fig. 1(b)). These features are complementary in that when one feature type is not reliably detected from a face image, the other feature(s) can compensate for it, in order to robustly estimate head pose. In each video frame, the system assesses the probability density function over the pose angle, ranging from -90^o to $+90^o$ (Fig. 1(b)). Spatial codebook representations are inferred from the various local features. For each feature type, we calculate the codebook statistics to infer the corresponding pose distribution. These are used in the graphical model to estimate the single video frame pose probability distribution. These head pose probabilities over the given video sequence, later, are temporally modelled. Finally, the non-parametric density estimation is employed to obtain fine discrete head pose probabilities. The results show that that the proposed framework outperforms competing methods [2, 4, 12, 40, 43] when evaluated on a challenging, unconstrained, public available video database, i.e. the McGillFaces Database [10] (see Fig. 1). The proposed model is also evaluated on the CMU Multi-PIE [14] database, which is collected in a controlled environment. It is observed that compared to the next closest competitor, our method achieves a much higher pose classication accuracy.

2 Methodology

In Level 1 (Fig. 3), the framework models the relationship between the statistics learned from different local invariant features on the detected face, and their corresponding face pose distributions. In Level 2, the single frame pose distribution is inferred based on the different feature-based face pose estimates inferred at Level 1. Finally, Level 3 (Fig. 2) estimates the most likely face pose configuration $\varTheta$ by leveraging the temporal information. To achieve, this we employ Belief Propagation (BP).

Assume that a video contains T video frames, each of which contains a successfully located and tracked face image via the algorithm in [10], $\boldsymbol{X^t_{int}}$, $t \in \{1, 2, \cdots, T\}$. Our goal is to estimate the set of head pose PDFs throughout the video, $\varTheta = \{\theta_1, \theta_2, \cdots, \theta_t, \cdots, \theta_T\}$ given $\boldsymbol{Y} = \{Y_1, Y_2, \cdots, Y_t, \cdots, Y_T\}$, where $\theta_t = \{\phi_1, \phi_2, \cdots, \phi_M\}$ is the set of head pose angles for each video frame, and $Y_t = \{y^t_{patch}, \ y^t_{edge}, y^t_{region}, y^t_{landmark}\}$, which are the patch, edge, region based and facial landmark based pose distributions. The posterior distribution is $p(\varTheta|\boldsymbol{Y}) = \frac{p(\varTheta, \boldsymbol{Y})}{p(\boldsymbol{Y})}$, where $p(\boldsymbol{Y})$ is a normalization term, which is constant with respect to $\varTheta$. To model the head pose over a video sequence $\varTheta$, the graphical model shown in Fig. 2 is employed. This allows us to express the posterior distribution with pairwise interactions: $p(\varTheta|\boldsymbol{Y}) = \frac{1}{p(\boldsymbol{Y})} \left(\prod_{t=1}^{T} \vartheta(\theta^t, Y^t)\right) \left(\prod_{t=1}^{T-1} \varphi(\theta^t, \theta^{t+1})\right)$. In this equation, the unary compatibility function accounting for local evidence (likelihood) for θ^t is represented by $\vartheta(\theta^t, Y^t)$, whereas the pairwise compatibility function between θ^t and θ^{t+1} is represented by $\varphi(\theta^t, \theta^{t+1})$.

The unary compatibility function for each node i, $\vartheta(\theta^t, Y^t)$, is defined as the joint distribution $p(\theta^t, Y^t)$ given by: $p(\theta^t, y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}) = p(\theta^t|y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})p(y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})$, where $p(\theta^t|y^t_{patch}, \ y^t_{edge}, y^t_{region}, \ y^t_{landmark})$ and $p(y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})$ are computed by a hierarchical graphical model, whose parametrization and learning are explained in the following subsections. A Gaussian distribution, $N(\mu, \varDelta)$, assumption is made to model the pairwise compatibility function $\varphi(\theta^t, \theta^{t+1})$.

Belief Propagation (BP) [29] is used to calculate the MAP estimate as the most likely head pose configuration, $\varTheta^* = \text{argmax}_{\varTheta} p(\varTheta|\boldsymbol{Y})$. In our experiments,

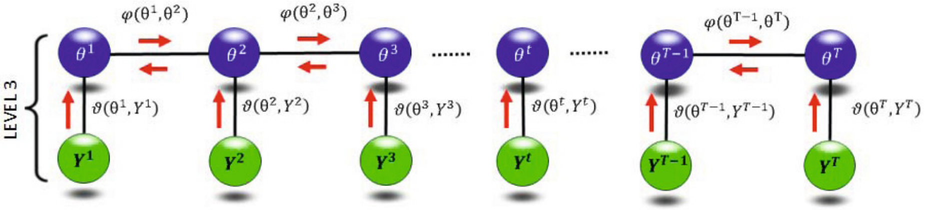

Fig. 2. Level 3 (the highest level) of the proposed framework: Belief Propagation defined over the proposed graphical model and local message passing for head pose estimation from a video sequence

we adapt the sum-product BP algorithm, which is efficient and provides the exact solution since the highest level of our model is a chain.

2.1 Hierarchical Temporal Graphical Model

$\boldsymbol{X^t_{int}}$ is the set of intensity values in RGB space at time t for the pixels in the image of the tracked face: $\boldsymbol{X^t_{int}} = \left\{x^t_{i,j} \mid \forall i \in \{1, \cdots, I\}, \forall j \in \{1, \cdots, I\}\right\}$, where the image has the size of $I \times I$ and $x^t_{i,j}$ is the intensity value of an individual pixel at location (i, j).

Given the challenges presented by real-world environments, local invariant features are inferred due to their high degree of robustness to various transformations, such as changes in scale, viewpoint, rotation and translation. Extensive analysis of a number of different local invariant features (e.g. [36, 37]) shows that using both densely and sparsely detected features, and representing these features with complementary descriptors is beneficial for classification/detection tasks.

The collection of the different feature representations (see Fig. 1(b)) inferred from the tracked face $\boldsymbol{X^t_{int}}$ is denoted by $\boldsymbol{X^t} = \left\{\boldsymbol{X^t_{patch}}, \boldsymbol{X^t_{edge}}, \boldsymbol{X^t_{region}}, \boldsymbol{X^t_{landmark}}\right\}$, where $\boldsymbol{X^t_{patch}}$ is the densely sampled patch representation, $\boldsymbol{X^t_{edge}}$ is the sparsely sampled edge representation, $\boldsymbol{X^t_{region}}$ is the dense region, and $\boldsymbol{X^t_{landmark}}$ is the facial landmark representation.

Here, $\boldsymbol{X^t_{patch}}$ is the collection of image patches extracted from $\boldsymbol{X^t_{int}}$:

$$\boldsymbol{X^t_{patch}} = \left\{x^t_p \mid \forall p \in \{1, \cdots P\}\right\}, \tag{1}$$

where P is the total number of patches and x^t_p denotes a single patch with index p, which contains two pieces of information: 1) the set of pixels in the patch, $\left\{x^t_{i,j}\right\}_p$, and 2) the location of the patch center, (r_p, c_p):

$$x^t_p = \left\{(\left\{x^t_{i,j}\right\}_p, r_p, c_p) \mid \left\{x^t_{i,j}\right\}_p \subset \boldsymbol{X^t_{int}},\ r_p \in \{1, \cdots, I\}, c_p \in \{1, \cdots, I\}\right\}. \tag{2}$$

$\boldsymbol{X^t_{edge}}$ denotes the collection of distinct points lying on the edge map inferred from X^t_{int}:

$$\boldsymbol{X^t_{edge}} = \left\{x^t_e \mid \forall e \in \{1, \cdots E\}\right\}, \tag{3}$$

where E is the total number of detected edge points, and x^t_e is a single edge point with edge index e, which contains two pieces of information: 1) the set of pixels $\left\{x^t_{i,j}\right\}_e$ that describes the e-th distinct edge point, and 2) location of the distinct edge point (r_e, c_e):

$$x^t_e = \left\{(\left\{x^t_{i,j}\right\}_e, r_e, c_e) \mid \left\{x^t_{i,j}\right\}_e \subset \boldsymbol{X^t_{int}},\ r_e \in \{1, \cdots, I\}, c_e \in \{1, \cdots, I\}\right\}. \tag{4}$$

$\boldsymbol{X^t_{region}}$ denotes the collection of facial regions extracted from X^t_{int}:

$$\boldsymbol{X^t_{region}} = \left\{x^t_r \mid \forall r \in \{1, \cdots R\}\right\}, \tag{5}$$

where R is the total number of facial regions extracted from $\boldsymbol{X^t_{int}}$, and x^t_r is the rth single face region, which includes three pieces of information: 1) its set of pixels $\left\{x^t_{i,j}\right\}_r$, 2) the location of the region center (r_r, c_r), and 3) the region scale (size) s_r:

$$\begin{aligned} x^t_r = \Big\{ & \left(\left\{x^t_{i,j}\right\}_r, r_r, c_r, s_r\right) \mid \left\{x^t_{i,j}\right\}_r \subset \boldsymbol{X^t_{int}}, \\ & r_r \in \{1, \cdots, I\}, c_r \in \{1, \cdots, I\}, s_r \in Z^+ \Big\}. \end{aligned} \tag{6}$$

$\boldsymbol{X^t_{landmark}}$ denotes the collection of facial landmarks extracted from X^t_{int}:

$$\boldsymbol{X^t_{landmark}} = \left\{x^t_{fl} \mid \forall l \in \{1, \cdots L\}\right\}, \tag{7}$$

where L is the total number of facial landmarks extracted from $\boldsymbol{X^t_{int}}$, and x^t_{fl} is the flth single facial landmark, which contains the facial landmark location (r_{fl}, c_{fl}).

To model each of these representations, one can use different features. Here, the features chosen are: (i) densely sampled "SIFT" [24] and "Color SIFT (CSIFT)" [9] features for modeling the face image patches, (ii) sparsely sampled "Geometric Blur (GB)" [5] features for modeling the distinct facial edge points, (iii) "Boundary Preserving Local Region (BPLR)" [21], and (iv) facial landmark [40] features for modeling the facial anatomical regions.

For the landmark features, we use the location information directly. For the remaining features, rather than using the pixel intensities directly for each feature type, the corresponding descriptor d is extracted from each feature point's intensity representation, such as $\left\{x^t_{i,j}\right\}_p$, $\left\{x^t_{i,j}\right\}_e$ or $\left\{x^t_{i,j}\right\}_r$. For the patch representation, SIFT and CSIFT descriptors are used, $d^t_{sift,p=k}$ and $d^t_{csift,p=k}$. In the case of edge features, the GB descriptor $d^t_{GB,e=l}$ is chosen. Pyramids of Histograms of Oriented Gradients are used as the region descriptor, i.e. $d^t_{PHOG,r=m}$. A visual vocabulary (codebook) is learned for each feature type, using the corresponding feature descriptors and an appropriate mapping function which takes the extracted feature's location information into account. That is, in the encoding step, hierarchical K-means clustering is performed on this information. Learning the optimal number of codewords is achieved via cross validation on training set. In the pooling step, vector quantization is used. Next, each extracted feature is represented by a visual word (codeword), from which codeword statistics will be learned. Occurrence statistics, for example, model how likely it is to observe a codeword for a pose value of interest. These statistics will be later used in the potential functions. Note that in the rest of the formulation, instead of the pixel intensity values, the corresponding descriptors d are used in $x^t_{p=k}$, $x^t_{e=l}$ and $x^t_{r=m}$.

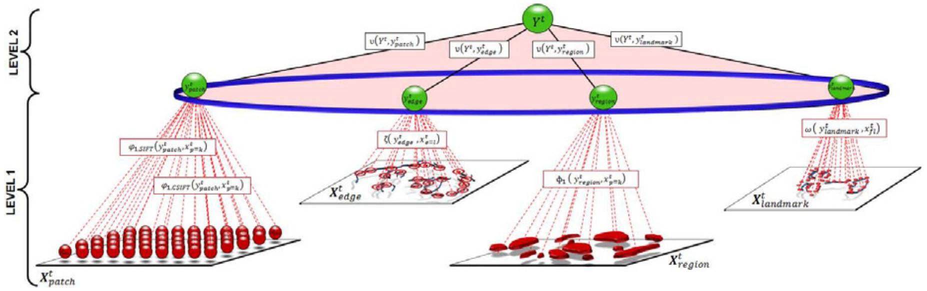
LEVEL 2
LEVEL 1
Y^t
$\upsilon(Y^t, y^t_{patch})$
$\upsilon(Y^t, y^t_{edge})$
$\upsilon(Y^t, y^t_{region})$
$\upsilon(Y^t, y^t_{landmark})$
y^t_{patch}
y^t_{edge}
y^t_{region}
$y^t_{landmark}$
$\varphi_{1,SIFT}(y^t_{patch}, x^t_{p=k})$
$\varphi_{1,CSIFT}(y^t_{patch}, x^t_{p=k})$
$\zeta(y^t_{edge}, x^t_{e=l})$
$\phi_1(y^t_{region}, x^t_{p=k})$
$\omega(y^t_{landmark}, x^t_{fl})$
X^t_{patch}
X^t_{edge}
X^t_{region}
$X^t_{landmark}$

Fig. 3. An overview of Level 1 and Level 2 of the hierarchical graphical model for the t-th frame in a video sequence. The red nodes are the patch, edge, region and landmark based visual words. The yellow nodes of y^t_{patch}, y^t_{edge}, y^t_{region} and $y^t_{landmark}$ represent pose distributions for the patch, edge, region and landmark representations, respectively. The green node Y^t is the pose distribution at the t-th frame. The boxes show the potential functions used to model the relationship between the corresponding two nodes.

We now define each of the facial feature types used in the proposed model: ***The Patch-based Representation:*** A dense sampling of the given facial image is achieved by the patch representation. SIFT and CSIFT vocabularies are used to model each image patch. This choice is motivated by the observed performance increase when CSIFT is combined with SIFT, as in [31]. To map the k-th image patch $x^t_{p=k}$ to a visual word $f^t_{sift,p=k}$ learned using the SIFT and CSIFT descriptors, a mapping function g is used such that $g : x^t_{p=k} \rightarrow f^t_{sift,p=k}$ and $g : x^t_{p=k} \rightarrow f^t_{csift,p=k}$ (that is $\boldsymbol{X^t_{patch}} \rightarrow \boldsymbol{F^t_{patch}}$). To leverage the spatial information, the patch location (r_p, c_p) is used in the coding and pooling phases, similar to the IG-BOW method in [10]. By adding two more dimensions to the descriptor space, this permits modeling the spatial inter-patch relationship. Because faces are aligned in the preprocessing step, this mapping provides better modeling for the face vocabulary. ***The Edge-based Representation:*** The Geometric Blur (GB) framework ([1, 5]) is used for detecting the key facial edge points and calculating the corresponding descriptor around each edge point. Geometric blur is shown to be effective when applied to sparse edge points. Thus, first the oriented edge filter responses are extracted from face images. Then, the rejection sampling over the edge map is used to obtain sparse interest points along edges. Once these interest points are detected, GB descriptors are calculated around each point [5]. GB descriptors, unlike uniform Gaussian blur-based descriptors, models the blur as small near the corresponding points, and larger far from them. Here the motivation is that the distortion due the affine transformations should be modeled properly: the amount of blur varies linearly with distance from corresponding points. To map the l-th distinct edge point $x^t_{e=l}$ to a visual word $f^t_{GB,e=l}$ learned using GB descriptors, we use the mapping function $g : x^t_{e=l} \rightarrow f^t_{GB,e=l}$ (that is $\boldsymbol{X^t_{edge}} \rightarrow \boldsymbol{F^t_{edge}}$). Similar to the facial patch occurrence model, the location information (r_e, c_e) is used in the coding and pooling steps. ***The Region-based Representation:*** To learn the pose information inherent in the facial anatomical regions, e.g the mouth, the eyes, the ears and the eyebrows, we use the boundary-preserving local regions (BPLRs) [21], i.e. $x^t_{r=m}$. Facial BPLRs are densely sampled local regions which preserve the shape of the facial structure on which they are detected. The BPLR detection is achieved based on the following steps [21]; The algorithm first obtains multiple overlapping segmentations from a given face image, for which the corresponding distance transform maps are computed. Then, it divides each segment into regular grid cells, and samples an element feature in each cell. The cell position and scale information is determined by the maximal distance transform value in the cell. Then, it links all elements using a minimum spanning tree, which extends connections and integrates multiple segmentations. Finally, it outputs a set of overlapping regions which contains a group of linked elements within the tree, namely BPLRs. Once the BPLRs are extracted, the Pyramids of Histograms of Oriented Gradients (PHOG) descriptors are computed over the gPb (globalized probability of boundary)-edge map for each detected BPLR. The mapping function $g : x^t_{r=m} \rightarrow f^t_{PHOG,r=m}$ not only uses the spatial information in the coding and pooling steps, but also the scale (size) information for each extracted BPLR. ***The Facial Landmark-based Representation:*** As facial landmarks [40] are shown to successfully estimate head pose when they are reliably detected, we also incorporate these features to our framework. The facial landmarks locations are used in the graphical model. Robust landmark extraction is achieved via the algorithm by Xiong and De la Torre [40].

2.2 Estimation of Level 2 Probabilities

The goal of this level is to estimate the posterior distribution $p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})$ for the face image in t-th video frame, by learning different combinations of patch, edge, region and landmark classifier pose distributions. To infer the posterior probability $p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})$, given the hierarchical model in Fig. 3, the following expression is proposed:

$$p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}) = \frac{1}{\mathcal{Z}(y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})} exp\left\{-U\right\}, \tag{8}$$

where as before, the normalization function is denoted by $\mathcal{Z}$ and U is the energy function defined as:

$$\begin{aligned}U = {} & \beta_1\nu(\theta^t, y^t_{patch}) + \beta_2\nu(\theta^t, y^t_{edge}) + \beta_3\nu(\theta^t, y^t_{region}) + \beta_4\nu(\theta^t, y^t_{landmark}) + \\ & \beta_5\nu(\theta^t, y^t_{patch}, y^t_{edge}) + \beta_6\nu(\theta^t, y^t_{patch}, y^t_{region}) + \beta_7\nu(\theta^t, y^t_{edge}, y^t_{region}) + ... \\ & \beta_{11}\nu(\theta^t, y^t_{patch}, y^t_{edge}, y^t_{region}) + ... + \beta_{15}\nu(\theta^t, y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}).\end{aligned} \tag{9}$$

$\{\beta_1, \cdots, \beta_{15}\}$ are the weights for each possible clique and potential function combinations, which are learned using the optimization strategy in [15], namely the de-randomized evolution strategy with covariance matrix adaptation (CMA-ES). CMA-ES is chosen since it does not require any prior information, such as the distribution shape, which is a difficult task considering the dimensionality of the β space. The potential function ν models the *unary*, *pairwise*, *triplet* and *fourth order* cliques of t-th frame pose distribution. Note that since we cannot show all 15 potential functions here, we show only a subset. The probability distribution functions are used to define corresponding ν :

$$\nu(\theta^t, y^t_{patch}) = -\log\left\{p(\theta^t | y^t_{patch}) p(y^t_{patch})\right\} \tag{10}$$

$$\nu(\theta^t, y^t_{patch}, y^t_{edge}) = -\log\left\{p(\theta^t | y^t_{patch}, y^t_{edge}) p(y^t_{patch}, y^t_{edge})\right\} \tag{11}$$

$$\begin{aligned}&\nu(\theta^t, y^t_{patch}, y^t_{edge}, y^t_{region}) \\ &= -\log\left\{p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region}) p(y^t_{patch}, y^t_{edge}, y^t_{region})\right\}\end{aligned} \tag{12}$$

$$\begin{aligned}&\nu(\theta^t, y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}) \\ &= -\log\left\{p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}) p(y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})\right\}.\end{aligned} \tag{13}$$

$\left\{y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}\right\}$ are the pose distributions, which are inferred through Section 2.3. The estimation of the joint probability $p(\theta^t, y^t_{patch})$ is achieved using the training database: $p(\theta^t, y^t_{patch}) \propto k(\theta^t, y^t_{patch}) + d_t$. The count of the joint occurrence event (θ^t, y^t_{patch}) is represented by $k(\theta^t, y^t_{patch})$, and the Dirichlet regularization parameter d_t is used to compensate for the sparsity. Because a uniform prior is assumed, d_t is constant for all t. Note that probabilities in other cliques are calculated in a similar fashion. The RFs [8], on the other hand, are used to calculate the posterior probabilities, such as $p(\theta^t | y^t_{patch})$, $p(\theta^t | y^t_{patch}, y^t_{edge})$, $p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region})$ and $p(\theta^t | y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark})$. Next, Gaussian kernel-based model fitting is employed to estimate the pose density in the range $[-90^o, +90^o]$ with 1-degree intervals. The motivation behind using such a kernel-based method is that the initial pose densities do not follow any specific parametric distribution. Note that it is possible to get even much finer pose intervals, if needed.

2.3 Estimation of Level 1 Probabilities

To infer $\left\{y^t_{patch}, y^t_{edge}, y^t_{region}, y^t_{landmark}\right\}$, we need to model the posterior distributions $p(y^t_{patch} \mid \boldsymbol{X^t_{patch}})$, $p(y^t_{edge} \mid \boldsymbol{X^t_{edge}})$, $p(y^t_{region} \mid \boldsymbol{X^t_{region}})$ and $p(y^t_{landmark} \mid \boldsymbol{X^t_{landmark}})$. The posterior distribution for the patch-based features is given by:

$$p(y^t_{patch} \mid \boldsymbol{X^t_{patch}}) = \frac{1}{\mathcal{Z}(\boldsymbol{X^t_{patch}})} exp\left\{-U_{patch}\right\}, \tag{14}$$

where $\mathcal{Z}$ is the normalization function and, given the proposed graphical model, the energy function U is defined as:

$$U_{patch} = \lambda_1 \varphi_{1,sift}(y^t_{patch}, \boldsymbol{X^t_{patch}}) + \lambda_2 \sum_{k=1}^{P} \varphi_{1,csift}(y^t_{patch}, \boldsymbol{X^t_{patch}}), \tag{15}$$

where the *unary* potential φ_1 models the relationship between patch features (e.g., SIFT or CSIFT) and the t-th frame patch-based pose distribution. The weights $\{\lambda_1, \lambda_2\}$ are learned from the training data using 2-fold cross validation.

For edge-based features, the posterior distribution is defined as:

$$p(y^t_{edge} \mid \boldsymbol{X^t_{edge}}) = \frac{1}{\mathcal{Z}(\boldsymbol{X^t_{edge}})} exp\left\{-U_{edge}\right\}, \tag{16}$$

where $\mathcal{Z}$ is the normalization function and the energy function U_{edge} is defined as: $U_{edge} = \zeta(y^t_{edge}, \boldsymbol{X^t_{edge}})$, where ζ is the edge related *unary* potential function, which models the relationship between edge features and the t-th frame edge-based pose distribution

The region and landmark posterior distributions, i.e. $p(y^t_{region} \mid \boldsymbol{X^t_{region}})$ and $p(y^t_{landmark} \mid \boldsymbol{X^t_{landmark}})$, are modeled via the *unary* potential functions of $\Phi_1(Y^t_{region}, \boldsymbol{X^t_{region}})$ and $\omega(Y^t_{landmark}, \boldsymbol{X^t_{landmark}})$, in a similar fashion to the edge-based features.

Facial Patch Potentials: The following expressions are used to model the occurrence potential function for SIFT and CSIFT based vocabulary (recall that using the codebook mapping $g : \boldsymbol{X^t_{patch}} \rightarrow \boldsymbol{F^t_{patch}}$ for SIFT and CSIFT separately):

$$\varphi_{1,sift}(y^t_{patch}, \boldsymbol{X^t_{patch}}) = -log\left\{p(y^t_{patch} \mid \boldsymbol{F^t_{patch}}) p(\boldsymbol{F^t_{patch}})\right\}, \tag{17}$$

where $\varphi_{1,csift}(y^t_{patch}, \boldsymbol{X^t_{patch}})$ is calculated similarly and uniform priors are assumed . Any classifier can be user to model the posterior probabilities $p(y^t_{patch} \mid \boldsymbol{F^t_{patch}})$ for SIFT and CSIFT. In this work, we choose to use a Random Forest (RF) [8] to perform inference. A RF is a discriminative classier that consists of an ensemble of decision tree classifiers, where the final classification is determined by summing the votes cast by each individual tree. Due to random selection of subset of training data and features, contrary to traditional decision trees, RF is less prone to overfitting,. Also The RF classifier is computationally efficient and also provides probabilistic outputs.

Table 1. The mean and standard deviation statistics of pose classification accuracy and RMSE for the different pose classification approaches, which are averaged over all folds

Method	Accuracy (%)	RMSE
Aghajanian and Prince [2]	20.68 ± 3.55	> 40
BenAbdelkader [4]	54.04 ± 8.77	> 40
Demirkus et al. [12]	55.04 ± 6.53	> 40
Xiong and De la Torre [40]	58.41 ± 9.61	29.81 ± 7.73
Zhu and Ramanan [43]	59.64 ± 7.66	35.70 ± 7.48
Our Method	$\mathbf{79.02} \pm \mathbf{3.79}$	$\mathbf{12.41} \pm \mathbf{1.60}$

Facial Edge Potential: The following expressions model the facial edge potential function using the mapping defined earlier, i.e. $g : \boldsymbol{X^t_{edge}} \rightarrow \boldsymbol{F^t_{edge}}$:

$$\zeta\left(y^t_{edge}, \boldsymbol{X^t_{edge}}\right) = -\log\left\{p\left(y^t_{edge} \mid \boldsymbol{F^t_{edge}}\right) p\left(\boldsymbol{F^t_{edge}}\right)\right\}, \tag{18}$$

where $p(\boldsymbol{F^t_{edge}})$ is assumed to be uniform, and the posterior probability $p(y^t_{edge} \mid \boldsymbol{F^t_{edge}})$ is also estimated using a Random Forest classifier.

Facial Region and Landmark Potentials: Modeling the potential functions $\Phi_1\left(y^t_{region}, \boldsymbol{X^t_{region}}\right)$ and $\omega\left(y^t_{landmark}, \boldsymbol{X^t_{landmark}}\right)$ is achieved by using a similar method to estimate the edge-based potential functions. To estimate the posterior probabilities $p(y^t_{region} \mid \boldsymbol{F^t_{region}})$ and $p(y^t_{landmark} \mid \boldsymbol{F^t_{landmark}})$, a Random Forest classifier is used.

2.4 Experimental Results

We begin by testing the proposed method on a fully unconstrained video dataset, and compare it to the top performing methods. To this end, we chose to test it on the McGillFaces Database [10]. This freely-accessible public database consists of 18,000 real-world video frames captured from 60 different subjects. The videos exhibit wide variability in captured head poses, with 45% of the frames showing non-frontal head poses, with more than half of these having poses beyond 45^o (see Fig. 1(a)). Each frame in the database has a labeled head pose. This ground-truth pose label is obtained using the robust 2-stage labeling strategy introduced in [10]. This labeling strategy provides *pose distributions*, in the range $[-90^o, +90^o]$, which can be interpreted as a measure of the labelers' belief of the pose angle. This labeling scheme provides 9 different discrete pose labels computed using the MAP estimates of the pose distributions (see Fig. 1(a)). The competing approaches provide only discrete pose estimates rather than complete pose distributions. Hence the discrete labels are used as the ground truth when testing the alternative approaches. In all cases, the tracking algorithm described in [10] is used to locate and track the faces in each video.

The proposed graphical model is compared to: (i) Aghajanian and Princes probabilistic patch-based within-object classification framework [2], (ii) BenAbdelkaders supervised manifold-based approach which uses raw pixel intensity

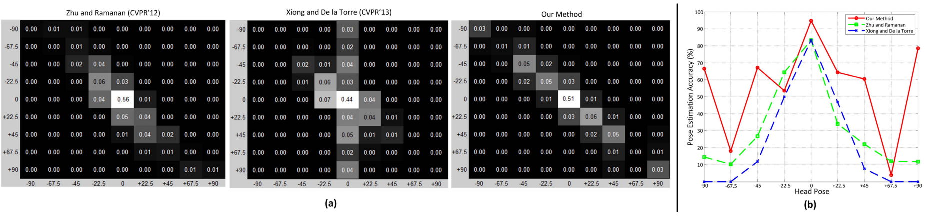

Fig. 4. Comparison of the proposed and the leading alternative methods: (a) the confusion matrices, and (b) the plot showing the pose estimation performance for each head pose label

values [4], (iii) Demirkus et al.'s Bayesian framework which uses OCI [33] features to model the pose in real-world environments [12], (iv) Zhu and Ramanan's unified model for face detection, pose estimation and landmark localization using a mixture of trees with a shared pool of parts [43], and (v) Xiong and De la Torre's Supervised Descent Method which solves a non-linear least squares problem in the context of facial feature detection and tracking (IntraFace) [40]. Ten-fold cross validation is used to evaluate the performance of each method, applied to videos taken from the McGillFaces Database.

In Table 1, quantitative comparison of the these approaches over all folds is provided. The validation metrics consist of: (1) head pose classification accuracy (results in terms of mean±std), and (2) the mean root mean square error (RMSE) based on angle error. In both categories, the proposed framework significantly (i.e., p-value of 4.9051×10^{-5} for the pose classification experiment compared to [43]) out-performs the alternative approaches. Our method provides the best accuracy and that the next closest competitors have over 19% lower accuracy. [43] and [40] are the bests among comparative approaches. [43] and [40] overall provide similar pose label estimation performance, whereas IntraFace [40] has a much lower RMSE. Note that the original implementation provided by the authors of [2, 40, 43] are optimized and used in our experiments. To accomplish a more comprehensive analysis, the confusion matrices (see Fig. 4(a)) and the plots showing the pose estimation performance for each head pose label (see Fig. 4(b)) are provided for the proposed model, [43] and [40]. The confusion matrix for [43] reveals that the approach is good at head pose estimation for face images depicting maximum of 45^o head pose angle. [43] does not provide reliable pose estimation for very off frontal, i.e. more than 67.5^o, face pose images. The confusion matrix for [40] shows that [40] is better at detecting faces in the wild however it has tendency to label face images with mostly frontal (in the range of $[-45^o,+45^o]$) pose label, leading to a dominant vertical flow in the confusion matrix. The confusion matrix for the proposed approach, on the other hand, shows a more diagonal trend with small variance. It is observed in Fig. 4(b) that all the methods, including the proposed method, perform poorly for pose angles of -67.5^o and $+67.5^o$. This is due to the fact that humans showed poor ability to perform ground truth labelling for these angles when shown images that are very unconstrained (see some failure cases in Fig. 5). The proposed

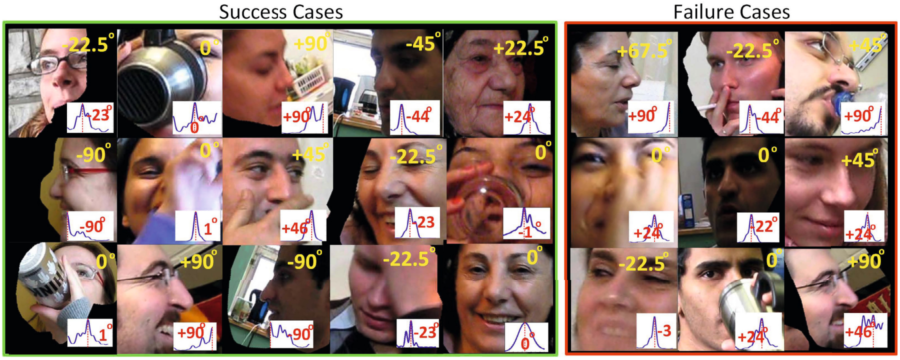

Fig. 5. Success and failure cases with the corresponding pose ground truth labels (in yellow), estimated pose distributions (in blue) and the MAP estimates (in red)

approach tends to estimate the label as being either -45^o or $+90^o$, for each of these angles respectively. As such, this reduced the overall reported accuracy in terms of bin classification. However, the RMSE errors for these angles helped this problem by reflecting the angle errors. Furthermore,the proposed approach is the only method to perform accurate pose estimation even for images acquired at full profile (90^o) poses. Furthermore, in Fig. 4(b), it is observed that all the three approaches do the best for the frontal images, which is expected. For [43] and [40], the performance decreases dramatically as the head pose is more off-frontal. The proposed approach, on the other hand, has a good pose estimation performance even for the images with full profile (90^o) poses.

The pose estimation performance for different combination of features is also analyzed. It is observed that the patch only, edge only, region only and landmark only achieves the pose estimation accuracy of 71.75%, 71.77%, 74.6%, 60.90%, respectively. That is, the region representation achieves the best performance among single features. When the region and patch features are combined, the performance increases to 76%. Combining the top three representations, i.e. region, patch and edge, leads to an accuracy of 77.13% whereas using all types of representations provides an accuracy of 79.02%. We also do a comparison of the MAP based accuracies for each pose label before and after the temporal stage. The maximum accuracy gain of 12.2% is achieved with a pose of -67.5^o, whereas the average accuracy gain over all pose bins is 9.17%. Over all folds, the mean histogram distance between the ground truth PDF and the estimated PDF decreases by 9.37% when Earth Mover's Distance is employed.

Fig. 5 shows examples of cases in which the proposed method does well, and where it fails. For each of these cases the estimated pose distributions are shown along with the MAP estimate of the head pose as well as the ground truth labels. One can also see that the method can be successful even in challenging conditions such as the presence of occlusion, facial hair and glasses, blur and various facial expressions. On the other hand, as depicted by some examples, the method can fail in the presence of motion blur, occlusions, due to the lack of reliable features.

Finally, we wish to examine the method under more controlled conditions. To this end, we test the proposed model on CMU Multi-PIE database [14], which is collected in a controlled environment. We perform a comparison on 5200 images which include all 13 different pose (equally distributed), along with lighting and facial expression variations. Our method achieves a pose classification accuracy of 94.46% whereas [43] provides an accuracy of 74.57%. Note that the pose classification is performed over 13 head pose bins and this dataset is larger than the one reported in [43].

3 Conclusions and Future Work

In this paper, we propose a hierarchical temporal graphical model to robustly estimate fine discrete head pose angle from real-world videos, i.e. with arbitrary facial expressions, arbitrary partial occlusions, arbitrary and non-uniform illumination conditions, motion blur and arbitrary background clutter. The proposed methodology provides a probability density function (pdf) over the range $[-90^o, +90^o]$ of head poses for each video frame rather than just provide a single decision. Experiments performed on the real-world video database (McGillFaces) and the controlled CMU Multi-PIE database show that the proposed approach significantly outperforms the alternative approaches. The proposed framework is a general approach which can be directly applied to any temporal trait and can use any type of feature. Our model does not rely on any subjective notion of what features are more useful for the task of interest. Rather, it learns how to optimally combine a set of features both spatially and temporally. It infers which set of features are more useful. Furthermore, the framework outputs the entire pose distribution for a given video frame, which permits robust temporal, probabilistic fusion of pose information over the entire video sequence. This also allows probabilistically embedding the head pose information for other tasks. We are currently collecting the probabilistic head pose ground truth for the YouTube Faces DB [38] to further evaluate our framework (the probabilistic labels will be publicly available). In future work, we plan to further analyze how temporal relationships can be used to improve other inference tasks (e.g. gender and facial hair classification).

References

1. Berg, A.C., Berg, T.L., Malik, J.: Shape matching and object recognition using low distortion correspondences. In: Proc. International Conference on Computer Vision and Pattern Recognition, CVPR (2005)
2. Aghajanian, J., Prince, S.: Face pose estimation in uncontrolled environments. In: Cavallaro, A., Prince, S., Alexander, D.C. (eds.) BMVC, pp. 1–11. British Machine Vision Association (2009)

3. Balasubramanian, V.N., Ye, J., Panchanathan, S.: Biased manifold embedding: A framework for person-independent head pose estimation. In: CVPR. IEEE Computer Society (2007)
4. BenAbdelkader, C.: Robust head pose estimation using supervised manifold learning. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part VI. LNCS, vol. 6316, pp. 518–531. Springer, Heidelberg (2010)
5. Berg, A.C., Malik, J.: Geometric blur for template matching. In: Proceedings of the 2001 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, pp. 607–614 (2001)
6. Beymer, D.: Face recognition under varying pose. In: Proceedings of the 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 1994, pp. 756–761 (June 1994)
7. Blanz, V., Grother, P., Phillips, P., Vetter, T.: Face recognition based on frontal views generated from non-frontal images. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 2, pp. 454–461 (June 2005)
8. Breiman, L.: Random forests. Machine Learning 45(1), 5–32 (2001)
9. Burghouts, G., Geusebroek, J.: Performance evaluation of local color invariants. Computer Vision and Image Understanding (CVIU) 113, 48–62 (2009)
10. Demirkus, M., Clark, J.J., Arbel, T.: Robust semi-automatic head pose labeling for real-world face video sequences. Multimedia Tools and Applications, 1–29 (2013)
11. Demirkus, M., Oreshkin, B.N., Clark, J.J., Arbel, T.: Spatial and probabilistic codebook template based head pose estimation from unconstrained environments. In: Macq, B., Schelkens, P. (eds.) ICIP, pp. 573–576. IEEE (2011)
12. Demirkus, M., Precup, D., Clark, J.J., Arbel, T.: Soft biometric trait classification from real-world face videos conditioned on head pose estimation. In: CVPR Workshops, pp. 130–137. IEEE (2012)
13. Demirkus, M., Precup, D., Clark, J.J., Arbel, T.: Multi-layer temporal graphical model for head pose estimation in real-world videos. In: ICIP. IEEE (2014)
14. Gross, R., Matthews, I., Cohn, J., Kanade, T., Baker, S.: Multi-pie. Image Vision Comput. 28(5), 807–813 (2010)
15. Hansen, N., Müller, S.D., Koumoutsakos, P.: Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). Evol. Comput. 11(1), 1–18 (2003)
16. Hassner, T.: Viewing real-world faces in 3D. In: The IEEE International Conference on Computer Vision, ICCV (December 2013)
17. Hu, C., Xiao, J., Matthews, I., Baker, S., Cohn, J.F., Kanade, T.: Fitting a single active appearance model simultaneously to multiple images. In: Hoppe, A., Barman, S., Ellis, T. (eds.) BMVC, pp. 1–10. BMVA Press (2004)
18. Hu, N., Huang, W., Ranganath, S.: Head pose estimation by non-linear embedding and mapping. In: ICIP (2), pp. 342–345. IEEE (2005)
19. Hua, G., Yang, M., Learned-Miller, E., Ma, Y., Turk, M., Kriegman, D., Huang, T.: Special section on real-world face recognition. IEEE Trans. Pattern Anal. Mach. Intell. 33(10), 1921–1924 (2011)
20. Huang, D., Storer, M., De la Torre, F., Bischof, H.: Supervised local subspace learning for continuous head pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2921–2928 (2011)

21. Kim, J., Grauman, K.: Boundary preserving dense local regions. In: Proc. International Conference on Computer Vision and Pattern Recognition, CVPR (2011)
22. Kumar, N., Berg, A., Belhumeur, P., Nayar, S.: Describable visual attributes for face verification and image search. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) 33(10), 1962–1977 (2011)
23. Li, H., Hua, G., Lin, Z., Brandt, J., Yang, J.: Probabilistic elastic matching for pose variant face verification. In: CVPR, pp. 3499–3506 (2013)
24. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision (IJCV) 60(2), 91–110 (2004)
25. Morency, L., Rahimi, A., Checka, N., Darrell, T.: Fast stereo-based head tracking for interactive environments. In: Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 390–395 (May 2002)
26. Murphy-Chutorian, E., Trivedi, M.M.: Head pose estimation in computer vision: A survey. IEEE Trans. Pattern Anal. Mach. Intell. 31(4), 607–626 (2009)
27. Oka, K., Sato, Y., Nakanishi, Y., Koike, H.: Head pose estimation system based on particle filtering with adaptive diffusion control. In: MVA, pp. 586–589 (2005)
28. Orozco, J., Gong, S., Xiang, T.: Head pose classification in crowded scenes. In: BMVC. British Machine Vision Association (2009)
29. Pearl, J.: Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufmann (1988)
30. Raytchev, B., Yoda, I., Sakaue, K.: Head pose estimation by nonlinear manifold learning. In: Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004, vol. 4, pp. 462–466 (August 2004)
31. Van de Sande, K.E.A., Gevers, T.: Evaluating color descriptors for object and scene recognition. IEEE PAMI 32(9), 1582–1596 (2010)
32. Sherrah, J., Gong, S.: Fusion of perceptual cues for robust tracking of head pose and position. Pattern Recognition 34(8), 1565–1572 (2001)
33. Toews, M., Arbel, T.: Detection, localization, and sex classification of faces from arbitrary viewpoints and under occlusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(9), 1567–1581 (2009)
34. Torki, M., Elgammal, A.: Regression from local features for viewpoint and pose estimation. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2603–2610 (November 2011)
35. Tosato, D., Farenzena, M., Spera, M., Murino, V., Cristani, M.: Multi-class classification on riemannian manifolds for video surveillance. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part II. LNCS, vol. 6312, pp. 378–391. Springer, Heidelberg (2010)
36. Tuytelaars, T.: Mikolajczyk, K.: Local invariant feature detectors: a survey. Foundations and Trends in Computer Graphics and Vision 3(3), pp. 177–280 (2008)
37. Tuytelaars, T., Mikolajczyk, K.: A survey on local invariant features. Tutorial at ECCV (2006)
38. Wolf, L., Hassner, T., Maoz, I.: Face recognition in unconstrained videos with matched background similarity. In: Proc. IEEE Conf. Comput. Vision Pattern Recognition (2011)
39. Wu, J., Trivedi, M.M.: A two-stage head pose estimation framework and evaluation. Pattern Recogn. 41(3), 1138–1158 (2008)

40. Xiong, X., De la Torre, F.: Supervised descent method and its application to face alignment. In: Proc. International Conference on Computer Vision and Pattern Recognition, CVPR (2013)
41. Yi, D., Lei, Z., Li, S.Z.: Towards pose robust face recognition. In: CVPR, pp. 3539–3545 (2013)
42. Zhao, G., Chen, L., Song, J., Chen, G.: Large head movement tracking using sift-based registration. In: Lienhart, R., Prasad, A.R., Hanjalic, A., Choi, S., Bailey, B.P., Sebe, N. (eds.) ACM Multimedia, pp. 807–810. ACM (2007)
43. Zhu, X., Ramanan, D.: Face detection, pose estimation, and landmark localization in the wild. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2879–2886 (June 2012)