Real-time head pose classification in uncontrolled environments with Spatio-Temporal Active Appearance Models

Miguel Reyes* and Sergio Escalera+ and Petia Radeva +

* *Matematica Aplicada i Analisi ,Universitat de Barcelona, Barcelona, Spain*
E-mail:mreyes@cvc.uab.es
+ *Matematica Aplicada i Analisi, Universitat de Barcelona, Barcelona, Spain*
E-mail:sergio@maia.ub.es
+ *Matematica Aplicada i Analisi, Universitat de Barcelona, Barcelona, Spain*
E-mail:petia@cvc.uab.es

Abstract

In this paper, we present a full-automatic real-time system for recognizing the head pose in uncontrolled environments over a cotinouos spatio-temporal behavior by the subject . The method is based on tracking facial features through Active Appearance Models. To differentiate and identify the different head pose we use a multi-classifier composed of different binary Support Vector Machines. Finally, we propose a continuous solution to the problem using the Tait-Bryan angles, addressing the problem of the head pose as an object that performs a rotary motion in dimensional space.

1 Introduction

Recognition of head pose has been a challenging problem in recent years, mainly due to interest in the technology industry to develop new interfaces to interact with the new generation of gadgets and applications. When we speak about head pose, we refer to its spatial location, treating the head like a point in a coordinate space, which can perform positional and rotational movements in any of the axes of the space where it moves. In most articles, the problem is addressed in a two-dimensional space, where a subject can make head movements on the horizontal and vertical axes. In a majority of published works[1] a two-stage sequential approach is adopted, aiming at locating the regions of interest and verifying the hypotheses on the head pose's presence (detection stage), and subsequently determining the type of the detected head pose (recognition stage).

In this paper, we perform a detection of a set of facial features. For both, detection and tracking, we base on Active Appearance Models. The face pose recovery is obtained by the analysis of image sequences from uncontrolled environments, performing detection and tracking based on continuous spatio-temporal constraints.

2 Method

Since the method should work in uncontrolled environments, we do not have information about the localization of the face. In this sense, we apply a windowing strategy using frontal and profile Viola-Jones [4] detectors, which provide positive bounding boxes for the initialization of face features location.

2.1 Detection and tracking of facial features

Once we have reduced the original image information into a region of interest where facial features are present, the next step is to fit all of them in a contextual model by means of a mesh using Active Appearance Models[2] (AAM). AMM benefit from Active Shape Models[3], providing information to the combination of shape and texture. Active Appearance Model is generated by combining a model of shape and texture variation. First, a set of points are marked on the face of the training images that are aligned, and a statistical shape model is built. Each training image is warped so the points match those of the mean shape. This is raster scanned in a texture vector, g, which is normalized by applying a linear transformation, $g \rightarrow \frac{(g-\mu_g 1)}{\sigma_g}$, where 1 is a vector of ones, and μ_g and σ_g^2 are the mean and variance of elements of g. After normalization $g_T 1 = 0$ and $|g| = 1$. Then, eigenanalysis is applied to build a texture model. Finally, the correlations between shape and texture are learnt to generate a combined appearance model. The appearance model has parameter c controlling the shape and texture according to:

$$x = \widehat{x} + Q_s c$$

$$g = \widehat{g} + Q_g c$$

where $\widehat{x}$ is the mean shape, $\widehat{g}$ the mean texture in a mean shaped patch, and Q_s, Q_g are matrices designing the modes of variation derived from the training set. A shape X in the image frame can be generated by applying a suitable transformation to the points, $x : X = St(x)$. Typically, St will be a similarity transformation described by a scaling s, an in-plane rotation, θ, and a translation (tx, ty). Once constructed the AAM, it is deformed on the image to detect and segment the face appearance as follows. During matching, we sample the pixels in the region of interest $g_{im} = T_u(g) = (u_1 + 1)gim + u_2 1$, where u is the vector of transformation parameters, and project into the texture model frame, $g_s = T_u^{-1}(g_{im})$. The current model texture is given by $g_m = \widehat{g} + Q_g c$. The current difference between model and image (measured in the normalized texture frame) is as follows:

$$r(p) = g_s - g_m$$

Given the error $E = |r|^2$, we compute the predict displacements $\delta p = -Rr(p)$, where $R = (\frac{\partial r^T}{\partial r}\frac{\partial p}{\partial p})^{-1}\frac{\partial r^T}{\partial p}$. The model parameters are updated $p \mapsto p + \kappa\delta$, where initially $\kappa = 1$. The new points and model $X^{'}$ frame texture $g_m^{'}$ are estimated, and the image is sampled at the new points to obtain g_{mi}, obtaining the new error vector as $r^{'} = T_{u'}^{-1} - g_m$. A final condition guides the end of each iteration: if $\left|r^{'}\right|^2 < E$, the we accept the new estimate, otherwise, we set to $\kappa = 0.5$, $\kappa = 0.25$, and so on. The procedure is repeated until no improvement is made to the error. Finally, we get a description of facial features through shape information ($X^{'}$) and texture (g_m). As for the performance of tracking, the acquisition of the features of shape and texture on the time $t + 1$, it is a slight variation of the facial features (shape and texture) on the previous time (t), therefore, the cost of convergence to the new pose is low, this is satisfied if met spatiotemporal behavior by the subject.

2.2 Head pose recovery

The description of facial features through AAM provides a vector descriptor of shape, consisting of 21 points which form the silhouette of a mesh. The target is to discretize the different types of mesh that can form, which will produce different head poses. We use a training set, labeling each point structure, and then perform discrete classification. The outputs of the classifier are as follows: right, middle-right, frontal, middle-left, and left. In order to get the discrete classifier, we establish a training set with different structures of points relating the head pose. These structures of points should be aligned. The support vector machine classifier[6] is used in a one-against-all design in

order to perform five-class classification. The final choice of the discrete output is done by majority voting. Taking into account the discontinuity that appears when a face moves from frontal to profile view we use three different AAM corresponding to three meshes: frontal view $\Im_F$, right lateral view $\Im_R$, left lateral view $\Im_L$. In order to include temporal and spatial coherence, meshes at frame $t+1$ are initialized by the fitted mesh points at frame t. Additionally. we include a temporal change-mesh control procedure, as follows

$$\Im^{t+1} = min_{\Im^{t+1}}\{E_{\Re_F}, E_{\Re_R}, E_{\Re_L}\}, \Im^{t+1}\epsilon\nu(\Im^t)$$

where $\nu(\Im^t)$ corresponds to the meshes contiguous to the mesh t fitted at time t (including the same mesh). This constraint avoids false jumps and imposes smoothness in the temporal face behavior (e.g. a jump from right to left profile view is not allowed).

In order to obtain a continuous output. The goal is to extract the angles of pitch and yaw movement between two consecutive frames. The angles are extracted from the following transformations:

$$R_{y,\theta} = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}$$

$$R_{x,\psi} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{pmatrix}$$

The transformation that will cause the motion is $R = R_{y,\theta}R_{x,\psi}$. Getting in V_i the information of shape in the frame t (trough AMM), and V_f relative to frame $t+1$, θ and ψ angles will be extracted by solving the following trigonometric equation:

$$RV_i = V_f$$

3 Results

First we clarify which data has been subjected for evaluation. Regarding the creation of the training set for the detection of facial features, we used

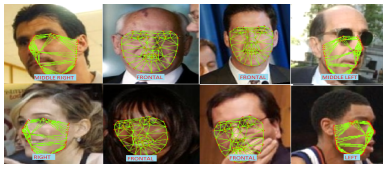

Figure 1: Discrete samples of the classifier output

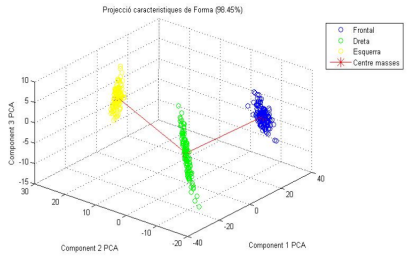

Figure 2: Shape projected on three PCA components

the database called Labeled Faces in the Wild[5]. This data set is composed of several samples from frontal, right, and left profiles. In Figure 2, shape characteristics of each of the classes are shown using 98.45% of principal components after manual labeling.

On the other hand, we get a representation of texture features as shown in Figure 3. Unlike the previous case, the projection of the characteristics of texture has a similar trend for all three classes. This is because the texture features are based mainly on skin color. Figure 3.

The next phase of testing is to evaluate the method on a video sequence. The results are shown in the next table.

Head poses % on a sequence of 7269 frames		
Head	% of Occurrence	System success
Left	0.1300	87
Middle-left	0.1470	73
Frontal	0.2940	95
Middle-right	0.1650	76
Right	0.2340	89

Finally we elaborated different tests to observe the continuous system output for the same problem (Figure 4).

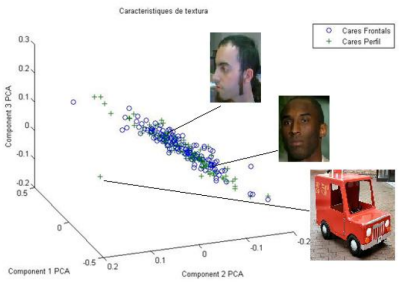

Figure 3: Texture projected on three PCA components

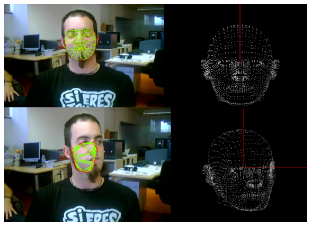

Figure 4: Initial frame $\phi = 0$ and $\theta = 0$ and final frame $\phi = 18.44$ and $\theta = -6.0385$

4 Conclusion

It is noted that AAM is able to detect a high number of facial features from different views in a robust and reliable way. Another quality that AAM is its ability to work with success in different situations from uncontrolled environments, such as partial occlusion or noise. It has also been observed that the robustness and reliability of facial feature extraction is very dependent on the number of points of the model. A greater number of points a more difficult to detect and track. Another important factor regarding the number of points that form a mesh is its computational cost, since this is entirely dependent on the density of dots forming a pattern. This is of paramount importance for real-time applications, since a high number of points can make the system unsustainable. In our work, a number of points greater than 21 can lead to problems of a slowdown in existing home computers.

Referring to the discrete outputs obtained by the classifier, we observed that the results were successful when the "head pose was near-fully frontal or profile". For the continuous output of the system, its success rate is totally dependent and sensitive to the monitoring carried out by AAM. Actually, the computational cost of calculating the angle difficulties the system to work in real time.

References

[1] Lisa M. Brown and Ying-Li Tian IBM T.J, "Comparative Study of Coarse Head Pose Estimation", *Watson Research Center Hawthorne, NY 10532*

[2] T. Cootes, J. Edwards, and C. Taylor, "Active Apperance Models. IEEE Transactions on Pattern Analysis and Machine Intelligence", *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(6):681685

[3] T. Cootes, C. Taylor, D. Cooper, and J. Graham, "Active Shape Models - their training and application", *Computer Vision and Image Understanding*, 61(1):3859.

[4] Paul Viola, Michael Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", *Conference on computer vision and pattern recognition 2001*,

[5] Gary B. Huang and Manu Ramesh and Tamara Berg and Erik Learned-Miller, "Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments", *University of Massachusetts, Amherst 2007*,

[6] David Meyer, Friedrich Leisch, and Kurt Hornik., "The support vector machine under test", *Neurocomputing 55(1-2): 169-186, 2003*,