ON THE EFFECT OF PERSPECTIVE DISTORTIONS IN FACE RECOGNITION

Zahid Riaz and Michael Beetz
Intelligent Autonomous Systems (IAS), Department of Computer Science,
Technical University of Munich, Munich, Germany

Keywords: Face Recognition, Active Appearance Models, Feature Extraction, Biometrics.

Abstract: Face recognition is one of the widely studied topics in the literature image processing and pattern recognition. Generally for the face images, the distance between camera and face is larger than the face size, hence in practice the effects of perspective distortions on the face edges are often ignored by the researchers. While these effects become more prominent if faces are viewed from different angles. In this paper, we study effects of perspective distortion and obtain improved results for face recognition against varying view-points. The approach follows by fitting a 3D model to the face images and creating a texture map by texture rectification at each triangle level. We compare our results with active appearance models (AAM) on two standard face databases.

1 INTRODUCTION

Over the past few years, several face recognition systems have been introduced in the commercial market and have been successfully applied under different scenarios (Zhao et al., 2003). Despite these efforts, the performance of the face recognition systems is still below iris and automated fingerprints identification systems (AFIS) due to uniqueness, permanence and reliability issues (Jain et al., 2005). However, on the other hand faces are natural way of interaction and non-intrusive. Cognitive sciences explains human face as a complex 3D object whose deformations can be controlled in a low dimensional subspace (O'Toole, 2009)(Blanz and Vetter, 2003) showing several meaningful variations which are challenging for the researchers towards the development of a face recognition system. These variations include changing view-points, varying lighting conditions and illuminations, facial expressions, temporal deformations like aging and image acquisition under different sessions of the day, occlusions and nevertheless spoofing.

In this paper we study face recognition problem under varying view-points and facial expressions by using a sparse 3D face model. Since faces are captured in actions, some of the facial areas are under self occlusions and considered as missing information. However, under slight out-of-plane rotations most of the facial areas are tilted and have very less textural information due to their oblique shape. In conventional face modeling approaches (Cootes et al., 1998), the distortions of these areas at the face edges are ignored by assuming that the distance between camera and the face is larger than the original face size. In these cases, effects of perspective distortions are ignored to obtain a high recognition rate. However, we study this problem in detail and obtain an improved performance by considering the effect of perspective distortion on local facial areas. The contributions of this paper are two fold: (1) to develop an unconstrained face recognition system which is robust against varying facial poses and slight facial expressions, (2) to study the effect of generally ignored perspective distortions on facial surface and recommend texture rectification. For this purpose, we use a generic 3D wireframe face model called Candide-III (Ahlberg, 2001). This model is defined with 184 triangular patches representing different areas on the surface of a 3D face. These are sparse flat triangles and capable to deform under facial action coding system (FACS) (Ekman and Friesen, 1978), action units and MPEG-4 facial animations units (Li and Jain, 2005). Each triangle defines a texture which is stored in a standard texture map by using camera rotation and translation. This rectified texture increases the recognition rate as compared to conventional 2D active appearance models (AAM) (Edwards et al.,

Riaz Z. and Beetz M..
ON THE EFFECT OF PERSPECTIVE DISTORTIONS IN FACE RECOGNITION.
DOI: 10.5220/0003859107180722
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2012), pages 718-722
ISBN: 978-989-8565-03-7

1998) on standard face databases. Model based approaches for human faces obtained popularity in last decade due to their compact and detailed representation (Blanz and Vetter, 2003)(Abate et al., 2007)(Park et al., 2004)(Cootes et al., 1998). In the literature of face modeling, some useful face models are point distribution models (PDM), 3D morphable models, photorealistic models, deformable models and wireframe models (Abate et al., 2007).

The remaining part of the paper is divided in three main section. Section 2 discusses face modeling in detail. In section 3, we thoroughly provide experiments performed using our approach as compared to conventional AAM. Finally section 4 gives conclusions of our work with future extension of this approach.

2 HUMAN FACE MODELING

We study structural and textural parameterization of the face model separately in this section.

2.1 Structural Modeling and Model Fitting

Our proposed algorithm is initialized by applying a face detector in the given image. We use Viola and Jones face detector (Viola and Jones, 2004). If a face is found then the system proceeds towards face model fitting. Structural features are obtained after fitting the model to the face image. For model fitting, local objective functions are calculated using haar-like features. An objective function is a cost function which is given by the equation 1. A fitting algorithm searches for the optimal parameters which minimizes the value of the objective function. For a given image I, if $E(I, c_i(\mathbf{p}))$ represents the magnitude of the edge at point $c_i(\mathbf{p})$, where $\mathbf{p}$ represents set of parameters describing the model, then objective function is given by:

$$f(I,\mathbf{p}) = \frac{1}{n}\sum_{i=1}^{n} f_i(I,c_i(\mathbf{p})) = \frac{1}{n}\sum_{i=1}^{n}(1 - E(I,c_i(\mathbf{p}))) \quad (1)$$

Where $n = 1, \ldots, 113$ is the number of vertices c_i describing the face model. This approach is less prone to errors because of better quality of annotated images which are provided to the system for training. Further, this approach is less laborious because the objective function design is replaced with automated learning. For details we refer to (Wimmer et al., 2008).

The geometry of the model is controlled by a set of action units and animation units. Any shape s can be written as a sum of mean shape $\bar{s}$ and a set of action units and shape units.

$$s(\alpha,\sigma) = \bar{s} + \phi_a\alpha + \phi_s\sigma \quad (2)$$

Where ϕ_a is the matrix of action unit vectors and ϕ_s is the matrix of shape vectors. Whereas α denotes action units parameters and σ denotes shape parameters (Li and Jain, 2005). Model deformation governs under facial action coding systems (FACS) principles (Ekman and Friesen, 1978). The scaling, rotation and translation of the model is described by

$$s(\alpha,\sigma,\pi) = mRs(\alpha,\sigma) + t \quad (3)$$

Where R and t are rotation and translation matrices respectively, m is the scaling factor and π contains six pose parameters plus a scaling factor. By changing the model parameters, it is possible to generate some global rotations and translations. We extract 85 parameters to control the structural deformation.

2.2 View Invariant Texture Extraction

The robustness of textural parameters depend upon the quality of input texture image. We consider perspective transformation because affine warping of the rendered triangle is not invariant to 3D rigid transformations. Affine warping works reasonably well if the triangle is not tilted with respect to the camera coordinate frame. However, most of the triangles on the edges are tilted and hence texture is heavily distorted in these triangles. In order to solve this problem we first apply perspective transformation. Since, the 3D position of each triangle vertex as well as the camera parameters are known, we determine the homogeneous mapping between the image plane and the texture coordinates by using homography H.

$$H = K.\begin{bmatrix} r_1 & r_2 & -R.t \end{bmatrix} \quad (4)$$

Where K, R and t denotes the camera matrix, rotation matrix and translation vector respectively, r_1 and r_2 are the components of rotation matrix. It maps a 2D point of the texture image to the corresponding 2D point of the rendered image of the triangle. A projection q of a general 3D point p in homogeneous coordinates is,

$$q = K.\begin{bmatrix} R & -R.t \end{bmatrix}.p \quad (5)$$

Each 3D homogeneous point lying on a plane with z = 0, i.e. p = (x y 0 1) leads to above equation. If p' being the homogeneous 2D point in texture coordinates then,

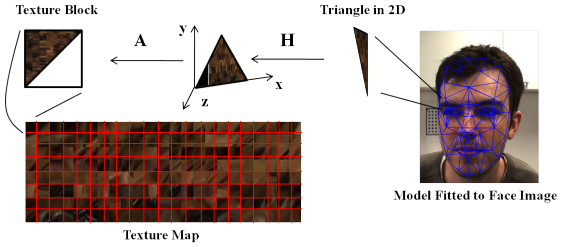

Figure 1: Detailed texture extraction approach described in section 2.2.

$$q = K.\begin{bmatrix} r_1 & r_2 & r_3 & -Rt \end{bmatrix}.\begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix} = H.p^{'}$$

Where R and t are the unknowns to be calculated. We use the upper triangle from each block of a rectangular texture map (see Figure 3) for storing the texture patches. In order to fit any arbitrary triangle to this upper triangle, we use an affine transformation A. The final homogeneous transformation M is given by,

$$M = A.K.\begin{bmatrix} R & -R.t \end{bmatrix} = A.H \tag{6}$$

The extracted texture vector g is parameterized by using principal component analysis (PCA) with mean vector g_m and matrix of eigenvectors P_g to obtain the parameter vector b_g (Li and Jain, 2005).

$$g = g_m + P_g b_g \tag{7}$$

2.3 Optimal Texture Representation

Each triangular patch represents meaningful texture which is stored in a square block of the texture map. A single unit of the texture map represents a triangular patch. We experiment with three different sizes of the texture blocks and choose an optimal size for our experimentation. These three block sizes include $2^3 \times 2^3$, $2^4 \times 2^4$ and $2^5 \times 2^5$. We calculate energy function from these texture maps of individual persons and observe the energy spectrum of the images in our database for each triangular patch. If N is the total number of images, and p_i be a texel value (which is equal to a single pixel value) in texture map, then we define energy function as:

$$E_j = \frac{1}{N}\sum_{i=1}^{N}(p_i - \overline{p}_j)^2 \tag{8}$$

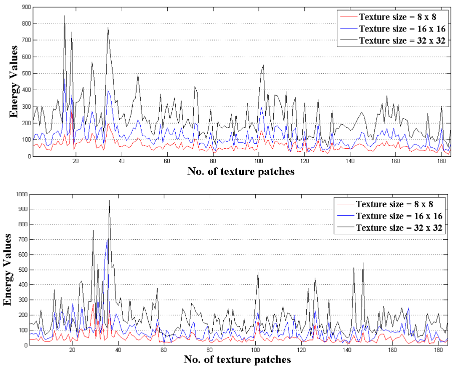

Figure 2: Energy spectrum of two randomly selected subjects from PIE database. Energy values for each patch is comparatively calculated and observed for three different texture sizes.

Where $\overline{p}_j$ is the mean value of the pixels in j^{th} block, $j = 1 \ldots M$ and $M = 184$ is the number of blocks in a texture map. In addition to Equation 8, we find variance energy by using PCA for each block and observe the energy spectrum. The variation within the given block has similar behavior for two kinds of energy functions except a slight variation in the energy values. Figure 2 shows the energy values for two different subjects randomly chosen from our experiments. It can be seen from Figure 2 that behavior of the textural components is similar between different texture sizes. The size of the raw feature vector extracted directly from texture map increases exponentially with the increase of texture block size. If $d \times d$ is the size of the block, then the length of the raw feature vector is $\frac{d(d+1)}{2}$. This vector length calculation depends upon how texture is stored in the texture map. This can be seen in Figure 3. We store each triangular patch from the face surface to upper triangle of the texture block. The size of raw feature vector extracted for $d = 2^3$, $d = 2^4$ and $d = 2^5$ is 6624, 25024 and 97152 respectively. Any higher value will exponentially increase the raw vector without any improvement in the texture energy. We do not consider higher values due to increase in vector length. The overall recognition rate produced by different texture sizes from eight randomly selected subjects with 2145 images from PIE database is shown in Figure 4. The results are obtained using decision trees and Bayesian networks for classification. The classification procedure is given in detail in next section 3. By trading off between the performance and size of the feature vectors, we choose texture block size to 16 × 16 during our experiments.

3 EXPERIMENTATION

In order to study perspective effects on face images, we experiment mainly on PIE-database (Ter-

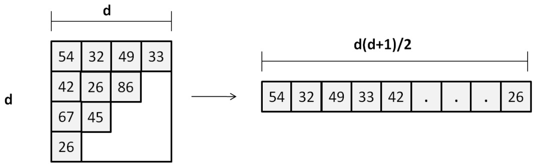

Figure 3: Texture from each triangular patch is stored as upper triangle of the texture block in texture map. A raw feature vector is obtained by concatenating the pixel values from each block.

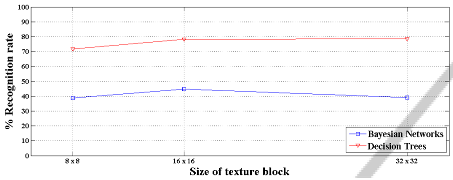

Figure 4: Comparison over eight random subjects from the database with three different sizes of texture blocks. Recognition rate slightly improved as texture size is increased however causes a high increase on the length of raw feature vector. We compromise on texture block of size 16 × 16.

ence et al., 2002) for face recognition and verify this fact on FG-NET (FG-NET, 2011) for age estimation. There are two sessions of PIE database captured from 1) October 2000 to November 2000 and 2) November 2000 to December 2000. The database contains pose, illuminations and facial expression variations. We experiment on all subject from second session of the database which consists of 15 subjects with 9162 images. Since our algorithm starts with Viola and Jones face detector (Viola and Jones, 2004), hence we consider only those images where face detector results are positive. In this filteration we obtain 3578 images where faces are successfully detected. The images with high pose variations (profile poses), dark effects and illuminations are filtered out in this step. We obtain images with frontal, half profile in both directions, looking upwards and looking downwards faces. Texture is extracted from each image by using method described in section 2.2. We obtain 3578 vectors of 25024 length. For dimensionality reduction, we use 40% of the data randomly selected from these raw features to learn PCA based subspace. For all texture sizes, we retain 97% of the covariance by choosing among the eigenvalues. The remaining data is projected on this space to obtain parameters which serve as feature vectors. The size of the parameter vector for three different textures is almost equal. This vector length is 188, 185 and 186 respectively for 8 × 8, 16 × 16 and 32 × 32.

For classification purpose, we apply decision tree. However, other classifiers can also be applied depending upon the application (Bayesian Networks (BN) were also used with comparable results during experimentation (refer Figure 4)). We choose J48 decision tree with 10-fold cross validation algorithm for experimentation which uses tree pruning called subtree raising and recursively classifies until the last leave is pure. The parameters used in decision tree are: confidence factor C = 0.25, with minimum two number of instances per leaf and C4.5 approach for reduced error-pruning (Witten and Frank, 2005). Face recognition rate under varying poses and facial expressions is given in Table 1. In order to verify the effect of perspective distortions, we further study age classification from all subjects of FG-NET database. This database consists of 1002 images of 62 subjects with age ranging from $0-69$ years. We divide the database in seven groups with 10 years band. The results are shown in Table 1.

Table 1: Comparison of traditional AAM approach and rectified texture. The results are shown for textural parameters and combined structural and textural parameters.

Database	**2D Texture parameters**	**Rectified Textural Parameters**	**AAM**	**3D Structural + Textural Parameters**
PIE	63.02%	**69.64%**	79.93%	**84.15%**
FG-NET	51.35%	**54.09%**	51.15%	**55.39%**

4 CONCLUSIONS AND FUTURE WORK

In this paper, we study a fact that performance of a face recognition system can be improved by considering the perspective effect on a face image. This issue is generally not given an attention by the research community. By ignoring this effect, better recognition rate can be achieved however by considering this effect further improvements are achieved. This approach gives equal weights to each triangular patch on the surface of the face. However, different weights can also be applied by considering the context and prior knowledge of the facial deformations. A remedy to texture rectification is proposed by applying a 3D model, which is sparse and achieves better recognition. The triangular patches which represent the face surface are flat, however curved triangles can further improve the results. Since, the proposed solution consists of 3D modeling, it is also recommended to use it for light modeling to obtain illumination invariance.

REFERENCES

Abate, A., Nappi, M., Riccio, D., and Sabatino, G. (2007). 2d and 3d face recognition: A survey. *Pattern Recognition Letters*, 28:1885–1906.

Ahlberg, J. (2001). An experiment on 3d face model adaptation using the active appearance algorithm. *Image Coding Group, Deptt of Electric Engineering, Linköping University*.

Blanz, V. and Vetter, T. (2003). Face recognition based on fitting a 3d morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1063–1074.

Cootes, T., Edwards, G., and Taylor, C. (1998). Active appearance models. *Proceedings of European Conference on Computer Vision*, 2:484–498.

Edwards, G., Cootes, T., and Taylor, C. J. (1998). Face recognition using active appearance models. In *5th European Conference on Computer Vision-Volume II*, ECCV '98, pages 581–595, London, UK. Springer-Verlag.

Ekman, P. and Friesen, W. (1978). The facial action coding system: A technique for the measurement of facial movement. *Consulting Psychologists Press*.

FG-NET (2011). Fg-net aging database. http://www.fgnet.rsunit.com/.

Jain, A. K., Bolle, R. M., and Pankanti, S. (2005). *Biometrics: Personal Identification in Networked Society*. Springer.

Li, S. Z. and Jain, A. K. (2005). *Handbook of Face Recognition*. Springer.

O'Toole, A. J. (2009). Cognitive and computational approaches to face recognition. *The University of Texas at Dallas*.

Park, I. K., Zhang, H., Vezhnevets, V., and Choh, H. (2004). H.k.: Image-based photorealistic 3-d face modeling. *In: FGR*, pages 49–56.

Terence, S., Baker, S., and Bsat, M. (2002). The cmu pose, illumination, and expression (pie) database. In *FGR '02: Fifth FGR*, page 53, Washington, DC, USA. IEEE Computer Society.

Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154.

Wimmer, M., Stulp, F., Pietzsch, S., and Radig, B. (2008). Learning local objective functions for robust face model fitting. *IEEE PAMI*, 30(8):1357–1370.

Witten, I. H. and Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, San Francisco.

Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey.