MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Pose Normalization via Learned 2D Warping for Fully Automatic Face Recognition

Asthana, A.; Jones, M.J.; Marks, T.K.; Tieu, K.; Goecke, R.

TR2011-055 August 2011

Abstract

We present a novel approach to pose-invariant face recognition that handles continuous pose variations, is not database-specific, and achieves high accuracy without any manual intervention. Our method uses multi dimensional Gaussian process regression to learn a nonlinear mapping function from the 2D shapes of faces at any non-frontal pose to the corresponding 2D frontal face shapes. We use this mapping to take an input image of a new face at an arbitrary pose and pose-normalize it, generating a synthetic frontal image of the face that is then used for recognition. Our fully automatic system for face recognition includes automatic methods for extracting 2D facial feature points and accurately estimating 3D head pose, and this information is used as input to the 2D pose-normalization algorithm. The current system can handle pose variation up to 45 degrees to the left or right (yaw angle) and up to 30 degrees up or down (pitch angle). The system demonstrates high accuracy in recognition experiments on the CMU-PIE, USF 3D, and Multi-PIE databases, showing excellent generalization across databases and convincingly outperforming other automatic methods.

British Machine Vision Conference (BMVC)

Pose Normalization via Learned 2D Warping for Fully Automatic Face Recognition

Akshay Asthana[1,2]
aasthana@rsise.anu.edu.au
Michael J. Jones[1] and Tim K. Marks[1]
{mjones,tmarks}@merl.com
Kinh H. Tieu[1,3]
kinh.tieu@gmail.com
Roland Goecke[2,4]
roland.goecke@ieee.org

[1]Mitsubishi Electric Research Labs
Cambridge, MA, USA
[2]CECS, Australian National University
Canberra, ACT, Australia
[3]now at Heartland Robotics
[4]Faculty of ISE, University of Canberra
Canberra, ACT, Australia

Abstract

We present a novel approach to pose-invariant face recognition that handles continuous pose variations, is not database-specific, and achieves high accuracy without any manual intervention. Our method uses multidimensional Gaussian process regression to learn a nonlinear mapping function from the 2D shapes of faces at any non-frontal pose to the corresponding 2D frontal face shapes. We use this mapping to take an input image of a new face at an arbitrary pose and pose-normalize it, generating a synthetic frontal image of the face that is then used for recognition. Our fully automatic system for face recognition includes automatic methods for extracting 2D facial feature points and accurately estimating 3D head pose, and this information is used as input to the 2D pose-normalization algorithm. The current system can handle pose variation up to 45 degrees to the left or right (yaw angle) and up to 30 degrees up or down (pitch angle). The system demonstrates high accuracy in recognition experiments on the CMU-PIE, USF 3D, and Multi-PIE databases, showing excellent generalization across databases and convincingly outperforming other automatic methods.

1 Introduction

Pose variation is one of the most crucial factors that limits the utility of current state-of-the-art face recognition systems. Previous methods for improving face recognition accuracy under pose variation include [2, 5, 6, 8, 10, 19]. All of these methods other than Sarfraz et al. [19] suffer from at least one of the following drawbacks: They (1) are not fully automatic, (2) do not allow for continuous pose variation, or (3) are database specific (use the same database for training and testing). In [5], a 3D morphable model is fit to a non-frontal face image to synthesize a frontal view that is then used for face recognition. This method is very slow and requires manual input for correct alignment. In [6], a pose-specific, patch-based locally linear mapping is learned between a set of non-frontal faces and the corresponding frontal faces. This method can only handle a discrete set of poses and also relies on manual labeling of facial landmark points. In [10], a single active appearance model (AAM) is used

to fit a non-frontal face, but this method also relies on manual labeling. In [2], a gallery augmentation approach was proposed that relied of generating several synthetic non-frontal images from the frontal gallery images. This method is limited to a discrete set of poses and requires manual labelling of facial landmark points. The method of [8] is also pose specific, requiring a set of prototype non-frontal face images that have the same pose as the input face. Sarfraz et al. [18, 19] present a fully automatic technique to handle pose variations in face recognition. Their method learns a linear mapping from the feature vectors of non-frontal faces to those of the corresponding frontal faces, but this assumption of linearity between the feature vectors is restrictive.

In this paper, we propose a novel 2D pose normalization method that not only removes the restrictions of all of these previous methods but also achieves better face recognition accuracy. The proposed 2D pose-normalization method (Section 2) uses multidimensional Gaussian process regression [15] to learn a nonlinear mapping function from the 2D shapes of faces at any non-frontal pose to the corresponding 2D frontal face shapes. Using this learned warping function, our system can take an image of a new face at an arbitrary pose and *pose-normalize* it, generating a synthetic frontal face by warping the non-frontal input image into a corresponding frontal shape. Unlike other 2D methods [2, 6] that can only handle a predetermined discrete set of poses, our method is designed to handle continuous pose variation accurately. Our fully automatic system includes robust methods for locating facial landmark points and estimating head pose accurately (Section 3). Finally, we show that our face recognition system outperforms state-of-the-art results on the CMU PIE, USF 3D, and Multi-PIE databases (Section 4).

2 2D Pose Normalization

The proposed method of 2D pose normalization is capable of generating a synthetic frontal face image from a single non-frontal image of a previously unseen face at any pose. It is based on using multidimensional Gaussian process regression [15] to learn a mapping between the 2D geometry of non-frontal faces at any poses and the corresponding frontal faces (Section 2.1). Given a non-frontal face image, our system first estimates the face's 2D shape and 3D pose (see Section 3). Using these estimates and the learned mapping, the system generates a predicted frontal shape and warps the texture from the non-frontal input shape to this predicted frontal shape via extended 2D piecewise-affine warping (Section 2.2). The ability of the proposed method to handle previously unseen poses is experimentally validated in Section 2.3.

Gaussian Process Regression: Throughout this paper, whenever we use Gaussian process regression [15], we use a squared exponential covariance function of the form

$$k(\mathbf{x}_p,\mathbf{x}_q) = \sigma_f^2 \exp\left(-\tfrac{1}{2}(\mathbf{x}_p-\mathbf{x}_q)^{\mathrm{T}}\mathbf{M}(\mathbf{x}_p-\mathbf{x}_q)\right) + \sigma_\varepsilon^2\delta_{pq}\,, \tag{1}$$

in which δ_{pq} is the Kronecker delta and $\mathbf{M}$ is a diagonal matrix, $\mathbf{M} = \mathrm{diag}([\lambda_1^{-2},\ldots,\lambda_d^{-2}])$, where d is the number of input dimensions. The hyperparameters to be learned are the signal variance, σ_f^2, the noise variance, σ_ε^2, and the characteristic length scale in each input dimension, $\lambda_1,\ldots,\lambda_d$. We use the conjugate gradient method to learn the values of the hyperparameters that maximize the marginal likelihood of the training data [15].

2.1 Continuous-Pose Regression (CPR)

Given training data for m subjects, consisting of the 2D shape of every subject's face at each of n poses (including the frontal pose), the goal here is to learn a regression function whose

input is the 2D shape of a new face at any pose (not necessarily one of the training poses) and whose output is the 2D shape of the same face in the frontal pose. Here, face shape is a vector containing the 2D locations of ℓ landmark points on the face. The main challenge is to learn a continuous regression function that can accurately predict the frontal face shape for an input face with arbitrary pose. This is a crucial requirement for a pose-invariant face recognition system in real-world scenarios, because in most probe images, the face's pose will not be in the discrete set of poses that were present in the training data.

Suppose we have training data at discrete poses P_i, where $i = \{1,\ldots,n\}$. For each pose P_i, the training data consist of the face shapes of every training subject, $\mathbf{s}_j^i$, where $j = \{1,\ldots,m\}$. The shape vector $\mathbf{s}_j^i$ contains the (x, y) coordinates of ℓ landmark points on the face. The goal is to learn a mapping function $\mathcal{F}$ from the shape vector of any face at any training pose to the corresponding shape vector at the frontal pose, P_1. Moreover, this function $\mathcal{F}$ should be continuous so that it can also map from any intermediate pose (not in the discrete set of training poses) to P_1.

We use Gaussian process regression [15] to learn this mapping function. Unlike previous regression-based methods [2, 16] that independently learn a separate regression function for each pose, we learn a multidimensional regression function $\mathcal{F}$ that applies to all poses, by including the 3D pose of the face as an input to the regression:

$$\mathcal{F} : \{\mathbf{s}_j^i, P_i\} \rightarrow \mathbf{s}_j^1 \,. \tag{2}$$

Our method for learning this warping function, which we call *Continuous-Pose Regression* (CPR), is explained in Algorithm 1. We learn $\mathcal{F}$ as a collection of independent functions F_h, each mapping a single landmark point's x or y coordinate from all poses to the frontal pose.

Algorithm 1: Training the Continuous-Pose Regression (CPR) Model

Require: Shape $\mathbf{s}_j^i$ for each subject $j \in \{1,\ldots,m\}$ and each pose $i \in \{1,\ldots,n\}$ in the training set,

$$\mathbf{s}_j^i = [x_1^{(i,j)} \;\; y_1^{(i,j)} \;\; \ldots \;\; x_\ell^{(i,j)} \;\; y_\ell^{(i,j)}],$$

where ℓ is the number of landmark points representing the face shape, and the pose $P_i = (\theta_i, \phi_i)$ is the 3D rotation of the face out of the image plane (θ is the yaw angle and ϕ is the pitch angle).

Goal : Learn CPR Model, $\mathcal{F}$, to predict a face's shape at pose P_1 from its shape at any pose P_i.

1 From the mn shape vectors $\mathbf{s}_j^i$, generate pairs of training samples $(\mathbf{T}_k, \mathbf{T}'_k)$, for $k \in \{1,\ldots,mn\}$, each representing the shape of a particular face j at pose $P_i = (\theta_i, \phi_i)$ and in the frontal pose P_1:

$$\mathbf{T}_k = \begin{bmatrix} x_1^{(i,j)} & y_1^{(i,j)} & \ldots & x_\ell^{(i,j)} & y_\ell^{(i,j)} \\ \theta_i & \theta_i & \ldots & \theta_i & \theta_i \\ \phi_i & \phi_i & \ldots & \phi_i & \phi_i \end{bmatrix}, \quad \mathbf{T}'_k = \begin{bmatrix} x_1^{(1,j)} & y_1^{(1,j)} & \ldots & x_\ell^{(1,j)} & y_\ell^{(1,j)} \end{bmatrix}. \tag{3}$$

2 Using Gaussian process regression [15], learn a collection $\mathcal{F}$ of regression functions F_h,

$$\mathcal{F} = \{F_1, F_2, \ldots, F_{2\ell}\}, \tag{4}$$

so that for each pair k of training samples and each column h of $\mathbf{T}_k$,

$$F_h(\mathbf{t}_{(k,h)}) \approx t'_{(k,h)}\,, \tag{5}$$

where $\mathbf{t}_{(k,h)}$ and $t'_{(k,h)}$ represent column h of $\mathbf{T}_k$ and of $\mathbf{T}'_k$, respectively.

2.2 Synthesizing a Frontal Image

Given a new non-frontal face image, our system estimates the face's 2D shape and 3D pose (see Section 3). Next comes pose normalization (Algorithm 2), in which the CPR model $\mathcal{F}$ (4) is used to predict the frontal face shape, and the texture from the given non-frontal image is warped to this predicted frontal shape via extended 2D piecewise-affine warping.

Piecewise-affine warping (PAW) is a commonly used 2D method for texture warping [2, 14]. Note that PAW can be used to warp the part of an image enclosed within the landmarks that make up a shape model. Our 68 landmark point annotation scheme, used throughout the experiments presented in this paper, does not cover some portions of the face near its outer boundary, such as the forehead, ears, and neck, but these regions may be important for recognition. In order to include these outer regions of the face in our warping, we make the approximation that pixels outside of (but nearby) our 68-point shape model share the same PAW parameters as the nearest PAW triangle. This *extended piecewise-affine warping* (extended PAW) enables us to warp features that lie outside of the annotated shape, thereby saving the extensive manual labor that would otherwise be required to label them separately.

Algorithm 2: Pose-Normalizing via CPR

Require: Non-frontal face image $\mathbf{I}$; Estimated shape $\mathbf{s}_{\text{test}}$ and estimated pose P_{test}:

$$\mathbf{s}_{\text{test}} = [x_1^{\text{test}} \;\; y_1^{\text{test}} \;\; \ldots \;\; x_\ell^{\text{test}} \;\; y_\ell^{\text{test}}] \qquad P_{\text{test}} = (\theta_{\text{test}}, \phi_{\text{test}}) \tag{6}$$

Goal : Generate pose-normalized frontal face image, $\mathbf{I}'$.

1 Generate a matrix $\mathbf{T}_{\text{test}}$ to be used as input to the collection of CPR regression functions, $\mathcal{F}$ (4):

$$\mathbf{T}_{\text{test}} = \begin{bmatrix} x_1^{\text{test}} & y_1^{\text{test}} & \ldots & x_\ell^{\text{test}} & y_\ell^{\text{test}} \\ \theta_{\text{test}} & \theta_{\text{test}} & \ldots & \theta_{\text{test}} & \theta_{\text{test}} \\ \phi_{\text{test}} & \phi_{\text{test}} & \ldots & \phi_{\text{test}} & \phi_{\text{test}} \end{bmatrix}. \tag{7}$$

2 Apply each regression function F_h (4) to $\mathbf{t}_h$ (column h of $\mathbf{T}_{\text{test}}$) to obtain t'_h (column h of the predicted frontal face shape, $\mathbf{s}'_{\text{test}}$):

$$\mathbf{s}'_{\text{test}} = [x'_1 \;\; y'_1 \;\; \ldots \;\; x'_\ell \;\; y'_\ell], \quad \text{where } t'_h = F_h(\mathbf{t}_h) \quad \text{for all } h \in \{1, \ldots, 2\ell\}. \tag{8}$$

3 Warp the texture of image $\mathbf{I}$ from the estimated shape $\mathbf{s}_{\text{test}}$ to the predicted shape $\mathbf{s}'_{\text{test}}$, using extended PAW, to obtain the pose-normalized image $\mathbf{I}'$.

2.3 Evaluating Pose Normalization on Trained and Untrained Poses

In this section, we test our CPR method's ability to generate the correct frontal face shapes from new non-frontal face shapes, both for poses that are in the discrete set of training poses and for poses that are not in the training set. We compare the performance of CPR to the shape regression method of [2]. That paper independently trained a separate Gaussian process regression function for each non-frontal pose, learning a mapping from the frontal face shapes to the corresponding non-frontal shapes. To compare that regression method with ours, we employ it in reverse, learning a mapping from face shapes in a specific non-frontal pose to the corresponding frontal face shapes. We refer to this method as *single-pose Gaussian process regression* (SP-GPR) because a separate regression is performed independently for each pose. Since SP-GPR is unable to handle poses that are not in the discrete set of training poses, we give SP-GPR training data for all poses, including those for which

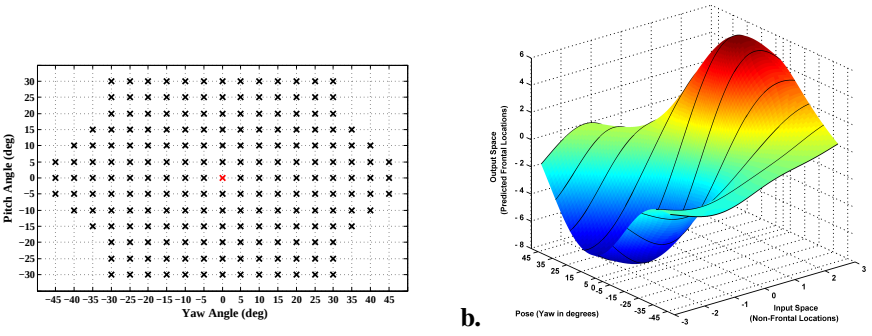

Figure 1: **a.** At every 3D pose (yaw and pitch angle) marked '×', we obtained 2D face shapes for all 100 subjects in the USF 3D database. **b.** Learned CPR function predicting the x-coordinate of the center of the upper lip on a frontal face (vertical axis) as a function of both the pose of the non-frontal face (yaw angle, lower left axis) and the x-coordinate of the center of the upper lip in that non-frontal pose (lower right axis). (For this graph, pitch angle is fixed at 0°, and axes representing the x-coordinates have been normalized.) Black lines on the surface denote the poses (yaw angles) used to train the function.

CPR has no training data. Despite this disadvantage, our CPR method performs as well as SP-GPR, even for the poses in which SP-GPR has training data but CPR does not.

Training and Testing Data: We obtained the 2D shapes (locations of 68 landmark points) for faces of 100 subjects across the 199 poses marked by '×' in Figure 1a, which cover up to ±45° in yaw and ±30° in pitch. To avoid the inordinate amount of time that would be required for carefully collecting and manually labeling so many images, we instead generated highly accurate 2D data for the 100 subjects in the USF Human ID 3D database [4]. Rather than hand-labeling each subject individually, we only needed to hand-label the mean 3D face at each pose, then use the one-to-one vertex correspondence among all subjects in the data set to propagate the vertex labels to all 100 subjects for that pose.

Experimental Validation: We illustrate the ability of CPR to handle unseen poses by using only the data with yaw angles of $0°, \pm 5°, \pm 15°, \pm 25°, \pm 35°, \pm 45°$ to train CPR. Thus, data from poses with yaw angles of $\pm 10°, \pm 20°, \pm 30°, \pm 40°$ are not trained, and can be used to test CPR's ability to handle continuous pose variation. The experiments follow a 2-fold cross-validation scheme where USF 3D data for 100 subjects are divided into 2 sets containing 50 subjects each. Figure 1b shows a 3D cross-section of the learned CPR model for the x-coordinate of one landmark point.

Figure 2 shows the RMSE (in pixels), averaged across all 100 subjects, between the frontal landmark positions predicted by CPR (pose-normalized faces) and the ground-truth frontal face landmark positions. Results are compared with those from SP-GPR [2] (trained separately for each individual pose) and a baseline comparison that simply uses the mean frontal face shape as the predicted frontal shape. Poses in Figure 2 indicate the yaw angle of the input face shapes (in this test, all input face shapes had 0° pitch). As shown in Figure 2b, even for each of the poses that the CPR model was not trained on, CPR performs as well as an SP-GPR model that was trained on only that one specific pose.

In the following sections, we show that 2D pose-normalization via CPR can be used effectively for pose-invariant face recognition in a fully automatic setup.

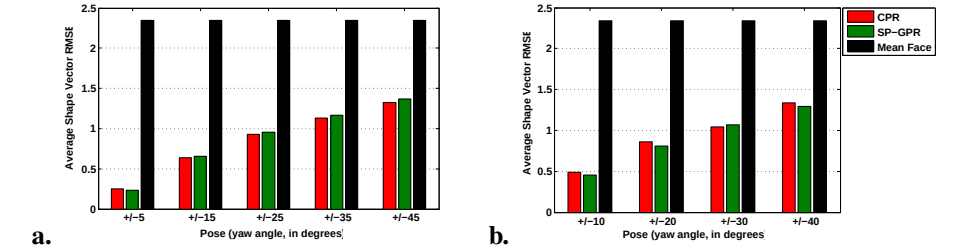

Figure 2: Average RMSE (in pixels) across all subjects on USF 3D data. Red bars show CPR error, green bars show SP-GPR error [7], and black bars show the error when the frontal shape is predicted as simply the mean frontal face shape. **a.** Poses on which both CPR and SP-GPR were trained. **b.** Poses on which SP-GPR was trained but CPR was not trained. Note the excellent generalization of CPR to previously unseen poses: Even for each pose on which it was not trained, CPR performs as well as an SP-GPR model that is specifically trained to handle just that single test pose.

3 Overview of our Fully Automatic System

Our fully automatic system for 2D pose normalization and face recognition is summarized in Figure 3. First, we run a multi-view face detector to find all faces in the input image. For each face, a set of facial feature detectors locates several facial landmark points, which are used to initialize a set of view-specific Active Appearance Models (VAAMs) [7] that are each fitted to the face image. The VAAM with the smallest fitting error is selected, and a global shape model is fit to this VAAM's 2D shape, yielding global shape parameters that are used to estimate 3D head pose (yaw and pitch). The pose information and the best-fitting VAAM points are used as input to the 2D pose-normalization system (Section 2), which outputs a synthetic frontal view of the face. The system presented in this paper covers poses up to $\pm 45^\circ$ yaw and $\pm 30^\circ$ pitch. More details are given below.

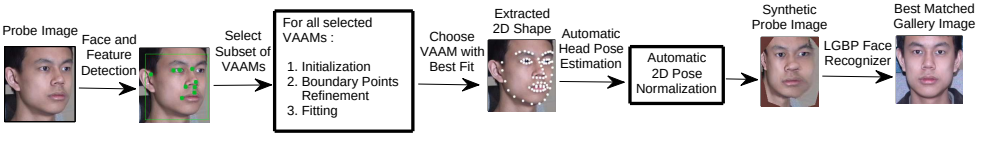

Figure 3: Overview of our fully automatic pose-invariant face recognition system.

Face and Feature Detection: Our system finds faces in the input image using a multi-view face detector, which consists of three separate Viola-Jones type detectors [21]: one each for left half-profile, frontal, and right half-profile faces. Together, these cover pose angles up to about $\pm 60^\circ$ yaw and $\pm 30^\circ$ pitch. Each of the 3 view-specific detectors is evaluated on a patch of the input image, classifying it as a face if any detector returns "face." For each face detected, the system runs a set of 9 view-specific feature detectors, which are Viola-Jones type detectors trained to detect specific points on the face such as eye corners and nose tip.
VAAM Initialization: Given the pose class estimated by the multi-view face detector (left half-profile, frontal, or right half-profile), a set of VAAMs is selected which covers that particular pose class. Each VAAM is initialized using the Procrustes method [11] to find the translation, in-plane rotation, and scale parameters that transform the mean shape of the VAAM to best match the locations of the detected feature points.

VAAM Point Refinement: If the pose class from the multi-view detector is left or right half-profile, the initial VAAM point locations are refined to ensure a better fit of the VAAM to the face boundary. Each VAAM point on the face boundary is allowed to move within a window around its initial location. We optimize the locations of all boundary points to maximize image gradient magnitude at the boundary, while minimizing the points' distances to their initial locations and changes in their positions relative to neighboring points. This process essentially fits a deformable face boundary model using dynamic programming [1].

Fitting via VAAMs: Next, each VAAM model is iteratively fit to the input image using the Efficient Approximation to the Simultaneous Inverse Compositional algorithm [3, 17], which optimizes both the shape and texture parameters of the VAAM. A shape error (distance of the fitted VAAM points from their initial locations) and a texture error (intensity difference between the pixels of the fitted VAAM model and the corresponding input image pixels) are computed. The best-fitting VAAM is the one with the smallest sum of shape and texture errors.

Automatic 3D Head Pose Estimation: A set of 68 facial landmark points is obtained from the best-fitting VAAM. Using these points, we normalize the roll angle of the face. Then, a global AAM (encompassing all poses covered by all the VAAMs) containing only shape components is fit to this normalized set of points. Since we are only using the global AAM shape model to estimate 2 parameters (yaw and pitch), an accurate shape is not needed, and we found that 5 parameters are enough to get a good pose estimate. The shape parameters of the global AAM are mapped to yaw and pitch angles using two separate Support Vector Regression (SVR) functions (one for yaw and one for pitch). These SVRs were trained by fitting the shape parameters of the global AAM to both ground truth and fitted VAAM points of the USF 3D [4] faces rendered at known pose angles.

Training VAAMs: The VAAMs were trained using data from the USF Human ID 3D database [4] and the Multi-PIE database [12]. From Multi-PIE, we used the data of 200 people in poses 05_1, 05_0, 04_1, 19_0, 14_0, 13_0, and 08_0 to capture the shape and texture variation induced by pose changes, and the data of 50 people in 18 illumination conditions to capture the texture variation induced by illumination changes. To extract the 2D shapes (68 landmark point locations) for all 100 subjects from the USF 3D database, the 3D mean face was hand labeled in 199 different poses (indicated by '×' in Figure 1a) to determine which 3D model vertex in each pose corresponds to each of the 68 landmark points. These vertex indices were then used to generate the 2D locations of all 68 points in each of the 199 poses for all 100 subjects in the USF 3D database. Generating 2D data from 3D models enables us to handle extreme poses in yaw and pitch accurately. This would not be possible using only 2D face databases for training, both because they do not have data for most of the poses marked in Figure 1a and because manual labeling would be required for each individual image. The VAAM shape models were trained on both the USF 3D and Multi-PIE data, but the VAAM texture models were trained using only the Multi-PIE data.

LGBP Face Recognizer: We have chosen to use the Local Gabor Binary Pattern (LGBP) recognizer [22]. The input to the recognizer is two face images that have been pose normalized as well as cropped and rectified to a canonical size. The output is a similarity score. Briefly, this recognizer works by computing histograms of oriented Gabor filter responses over a set of non-overlapping regions that tile an input image. The concatenation of all histogram bins forms a feature vector. Two feature vectors are compared by summing the histogram intersections over all the bins, which yields the similarity score. The LGBP recognizer has the advantage that there are no parameters to set, so no training is involved, and yet its performance is comparable to other state-of-the-art recognizers on many test sets.

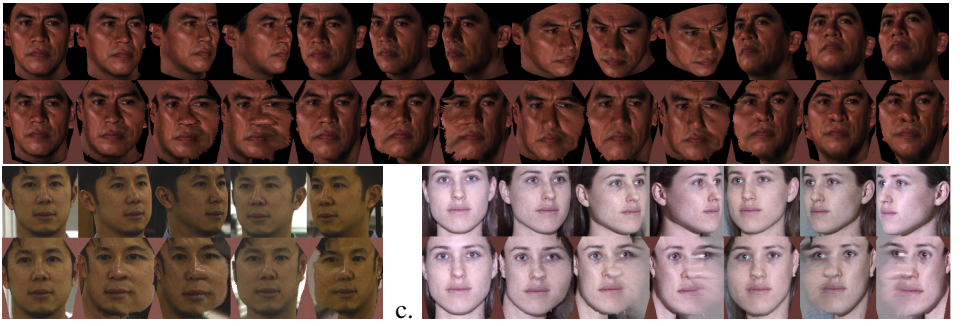

Figure 4: Examples of 2D Pose Normalization from (a) USF 3D, (b) CMU PIE, and (c) Multi-PIE. In each part, the top row contains example input images, with the corresponding pose-normalized images in the bottom row; the frontal gallery image is in the left column.

4 Experiments

CMU PIE											
Method	Alignment	Trained on PIE	Gallery Size	Poses Handled	c11	c29	c07	c09	c05	c37	**Avg**
Kanade03*[13]	manual	yes	34	discrete	96.8	100.0	100.0	100.0	100.0	100.0	99.5
Chai07* [6]	manual	no	68	discrete	89.8	100.0	98.7	98.7	98.5	82.6	94.7
Sarfraz10*[15, 19]	automatic	yes	34	continuous	87.9	89.2	99.8	92.8	91.5	87.9	91.5
Sarfraz10*[15, 19]	automatic	no	68	continuous	84.0	87.0	–	–	94.0	90.0	88.8
LGBP [22]	automatic	no	67	N/A	71.6	87.9	78.8	93.9	86.4	74.6	82.2
Ours	automatic	no	67	continuous	88.1	100.0	98.5	98.5	95.5	89.4	95.0

USF				
Pitch Range (°)	-15 to +15		-30 to -20 and +20 to +30	
Yaw Range (°)	LGBP	**Ours**	LGBP	**Ours**
-15 to +15	97.1	99.8	84.4	98.7
-30 to -20 and +20 to +30	88.8	98.8	67.2	96.3
-45 to -35 and +35 to +45	78.3	95.2	–	–

Multi-PIE								
Method	080_05 -45°	130_06 -30°	140_06 -15°	051_07 0°	050_08 +15°	041_08 +30°	190_08 +45°	**Avg**
LGBP [22]	37.7	62.7	77.0	92.6	83.0	58.7	35.9	62.0
Ours	43.8	83.3	94.0	96.3	94.7	70.0	41.2	74.8

Table 1: Pose-wise rank-1 recognition rates (in %) for CMU PIE, USF 3D, and Multi-PIE databases. The numbers for the starred(*) methods were estimated from plots in [6, 13, 15]. To get LGBP baseline results, we first performed 2D alignment using our automatic feature detectors, then used code from the authors of [22].

We conducted face recognition experiments using the USF Human ID 3D [4], CMU PIE [20], and Multi-PIE [12] databases. Figure 4 shows pose normalization examples. The CMU PIE database has been used by many other researchers to test face recognition across pose, which allows us to compare with previous approaches. We can generate faces at any pose using the USF data set, which allows us to demonstrate our system's ability to handle both yaw and pitch variations simultaneously. Multi-PIE is the most recent and the most challenging of the data sets, and our experiments on it set a baseline for future comparisons.

Given a test image, our system automatically detects the face and facial landmarks that

are used to initialize the 2D VAAMs (Section 3). A Failure To Acquire (FTA) occurs if the face is not detected or fewer than 3 facial features are located, in which case no pose-normalized face image is output. For all other images, the selected 2D VAAMs are fit to the face to find the optimal locations of the 2D VAAM landmark points. These are used to compute the global shape parameters that are used to estimate the 3D head pose. This head pose information and the 2D VAAM landmark locations are used by our 2D pose-normalization system (Section 2) to generate the synthetic frontal face image that is passed to the LGPB face recognizer for comparison with all of the gallery images. Our system requires no manual intervention and acheives robust pose-invariant face recognition. The entire 2D pose-normalization process takes about 6 seconds on a modern Pentium processor.

In our recognition experiments, we remove any FTA cases from the test set so they are not counted as recognition errors. Any automatic system has the issue of FTAs. We report the percentage of FTA cases for each test set below. Doing this allows us to distinguish clearly between detection failures and recognition failures.

USF 3D Database: For the USF Human ID 3D database [4], we rendered 199 different poses (Figure 1a) up to $\pm 45^\circ$ yaw and $\pm 30^\circ$ pitch for each of the 94 unique subjects. The gallery set consisted of the frontal images of each subject (94 total). The remaining 18,612 images formed the probe set. The FTA rate was 3.37%, which is higher than on the other data sets due to the combined large yaw and pitch angles of the probes. Our method obtained a **97.8%** rank-1 recognition rate overall on this test set. Table 1 shows the recognition rates broken down by pose class.

CMU PIE Database: For this test set, the gallery consisted of the frontal image (Pose ID c27) with neutral expression and ambient lighting for each of the 68 subjects. The probe set consisted of 6 non-frontal poses (see Table 1) also with neutral expression and ambient lighting for each subject. The gallery image for one subject (Subject ID 04021) was an FTA case for our face detector. Since no cropped gallery image could be obtained, we removed that subject from our results. The remaining 67 subjects were used in our recognition test, which had a 1.1% FTA rate. Our method's overall rank-1 recognition rate on this set was **95.0%**. Table 1 compares our results (broken down by pose class) to previous methods.

Multi-PIE Database: For this test set, we used 137 subjects (Subject ID 201–346) with neutral expression at 7 different poses from all 4 sessions, with illumination that is frontal with respect to the face pose (see Table 1). Our VAAM model was trained on 200 Multi-PIE subjects (Subject ID 001–200), who were not used in our recognition test set. The gallery set consisted of the frontal image from the earliest session for each test subject (137 total). The probe set contained all of the remaining images per subject including frontal images from other sessions (1,963 total). The FTA rate was 1.2%. Our method obtained a **74.8%** overall rank-1 recognition rate on this test set. Table 1 shows recognition rates for each pose class.

Summary of Results: The results show that our method improves upon the state of the art for data sets with wide pose variation. Unlike most previous methods, our system is fully automatic, handles continuous pose, and generalizes well to data sets on which it has not been trained. The system described by Sarfraz et al. [18, 19] also overcomes these limitations but has significantly worse results on the CMU PIE test set. Furthermore, on each test set, our results using pose normalization are significantly better than the LGBP baseline results with no pose normalization. The relatively poor performance at the $\pm 45^\circ$ poses of Multi-PIE is most likely due to the fact that, unlike in the other test sets, the $\pm 45^\circ$ Multi-PIE images have almost half of the face occluded due to the out-of-plane rotation, making it a particularly challenging data set.

5 Conclusion

Our pose-invariant face recognition system is fully automatic, accommodates a wide range of poses, and achieves state-of-the-art performance. Our method is based on fitting 2D models, which makes it very computationally efficient compared to 3D model-based methods [5]. Another advantage of our approach is that it handles continuous pose variations, unlike some previous methods [2, 6] that limit probe images to a fixed set of discrete poses. In the future, we plan to extend our system to a wider range of poses. One difficulty will be that 2D texture warping methods perform poorly in the presence of large self-occlusions, so additional techniques such as texture synthesis or inpainting may be needed.

References

[1] A. A. Amini, T. E. Weymouth, and R. C. Jain. Using dynamic programming for solving variational problems in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 12:855–867, September 1990.

[2] A. Asthana, C. Sanderson, T. Gedeon, and R. Goecke. Learning-based face synthesis for pose-robust recognition from single image. In *British Machine Vision Conference (BMVC)*, 2009.

[3] S. Baker, R. Gross, and I. Matthews. Lucas-Kanade 20 years on: A unifying framework: Part 3. Technical report, Robotics Institute, Carnegie Mellon University, USA, 2003.

[4] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *ACM SIGGRAPH*. 1999.

[5] V. Blanz and T. Vetter. Face Recognition Based on Fitting a 3D Morphable Model. *IEEE Transaction on Pattern Analysis and Machine Intelligence (PAMI)*, 25(9):1063–1074, September 2003.

[6] X. Chai, S. Shan, X. Chen, and W. Gao. Locally linear regression for pose-invariant face recognition. *IEEE Transactions in Image Processing*, 16(7):1716 –1725, 2007.

[7] T.F. Cootes, K.N. Walker, and C.J.Taylor. View-Based Active Appearance Models. In *IEEE FG*, 2000.

[8] S. Du and R. Ward. Component-wise pose normalization for pose-invariant face recognition. In *IEEE ICASSP*, 2009.

[9] G. Edwards, C.J. Taylor, and T.F. Cootes. Interpreting Face Images Using Active Appearance Models. In *IEEE FG*, 1998.

[10] H. Gao, H. K. Ekenel, and R. Stiefelhagen. Pose normalization for local appearance-based face recognition. In *ICB*, 2009.

[11] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 1989.

[12] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. In *IEEE FG*, 2008.

[13] T. Kanade and A. Yamada. Multi-Subregion Based Probabilistic Approach Toward Pose-Invariant Face Recognition. In *IEEE CIRA*, 2003.

[14] I. Matthews and S. Baker. Active Appearance Models Revisited. *International Journal of Computer Vision (IJCV)*, 60(2):135–164, 2004.

[15] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006.

[16] O. Rudovic, I. Patras, and M. Pantic. Coupled gaussian process regression for pose-invariant facial expression recognition. In *ECCV*. 2010.

[17] J. M. Saragih and R. Goecke. Learning AAM fitting through simulation. *Pattern Recognition*, 42(11):2628–2636, November 2009.

[18] M. S. Sarfraz. *Towards Automatic Face Recognition in Unconstrained Scenarios*. PhD thesis, Technische Universität Berlin, 2008.

[19] M. S. Sarfraz and O. Hellwich. Probabilistic learning for fully automatic face recognition across pose. *Image Vision Computing (IVC)*, 28:744–753, May 2010.

[20] T. Sim, S. Baker, and M. Bsat. The CMU Pose, Illumination, and Expression Database. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 25(12): 1615–1618, 2003.

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

[22] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition. *IEEE ICCV*, 2005.