Shape-constrained Gaussian Process Regression for Facial-point-based Head-pose Normalization

Ognjen Rudovic[1] and Maja Pantic[1,2]

[1]Comp. Dept, Imperial College London, UK
[2]EEMCS, University of Twente, The Netherlands
{o.rudovic,m.pantic}@imperial.ac.uk

Abstract

Given the facial points extracted from an image of a face in an arbitrary pose, the goal of facial-point-based head-pose normalization is to obtain the corresponding facial points in a predefined pose (e.g., frontal). This involves inference of complex and high-dimensional mappings due to the large number of the facial points employed, and due to differences in head-pose and facial expression. Most regression-based approaches for learning such mappings focus on modeling correlations only between the inputs (i.e., the facial points in a non-frontal pose) and the outputs (i.e., the facial points in the frontal pose), but not within the inputs and the outputs of the model. This makes these models prone to errors due to noise and outliers in test data, often resulting in anatomically impossible facial configurations formed by their predictions. To address this, we propose Shape-constrained Gaussian Process (SC-GP) regression for facial-point-based head-pose normalization. Specifically, a deformable face-shape model is used to learn a face-shape prior, which is placed on both the input and the output of GP regression in order to constrain the model predictions to anatomically feasible facial configurations. Our extensive experiments on both synthetic and real image data show that the proposed approach generalizes well across poses and handles successfully noise and outliers in test data. In addition, the proposed model outperforms previously proposed approaches to facial-point-based head-pose normalization.

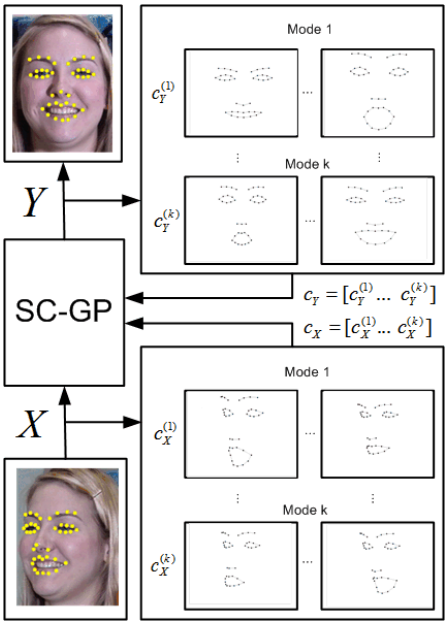

Figure 1. **Outline of the method:** SC-GP regression maps the locations of 39 facial landmark points (X) extracted from a facial image in a non-frontal view to the corresponding points (Y) in the frontal pose by means of the structured prediction based on deformable shape models (learned independently for the input and output, and described by the first k deformable modes).

1. Introduction

Head-pose normalization (i.e., canceling the effect of head rotation) is a crucial pre-processing step for computer-vision applications such as person identification and facial behaviour analysis. These, in turn, are an integral part of many real-world technologies including smart-rooms interfaces, user-adaptive tutoring systems, etc., where the assumption of having non-movable subjects is unrealistic. However, due to differences in head-pose, large variation in appearance of facial expressions across different poses, and difficulty in decoupling these two sources of variation, head-pose normalization of expressive facial data poses a significant research challenge.

The main aim of facial-point-based head-pose normalization is to generate a 'virtual' pose, that is, to normalize facial points localized in the input image by mapping them

to the corresponding facial points in a predefined pose (e.g., frontal), before further analysis is conducted. A common approach to this problem in computer vision is to use a 3D or 2D face-shape model. Blanz et al. [3] proposed a morphable model to reconstruct a 3D face-shape from an input image, based on seven facial points. Wang and Lien [18] applied an affine transformation to learn back-projection from locations of twenty-one facial points, localized in the input image, to a 3D virtual face model. Zhu and Ji [19] proposed a normalized singular value decomposition algorithm to separate head pose from facial expressions through parameterization of a 3D-Point Distribution Model (PDM), based on twenty-eight 2D facial points. Dornaika and Orozco [8] proposed an online Active Appearance Model (AAM), with the 3D Candide model (as the underlying 3D-Active Shape Model (ASM)), which encodes information about head pose and facial actions. In general, all these methods can be used to achieve facial-point-based head-pose normalization: the 3D face-shape, estimated from 2D facial points in an arbitrary pose, is first rotated to the frontal pose, and then the target 2D points in the frontal pose are obtained from the 3D face-shape. Nevertheless, estimating 3D face-shape from 2D facial points is by no means a trivial task since it involves inference of complex, high-dimensional, and multimodal mappings. Moreover, these models employ computationally expensive fitting techniques, based on gradient optimization algorithms, which may fail to converge, hence, resulting in an inaccurate 3D face-shape [8]. This, in turn, will inevitably impair the head-pose-normalization achieved by these models.

The 2D face-shape-based methods for head-pose-normalization have also been proposed [6, 9]. Cootes et al. [6] proposed pose-based AAMs, where the face-shapes in each pose are learned by means of 2D-PDM, based on the sixty-five facial points. The facial points in the frontal pose are obtained by mapping non-frontal face-shapes to the corresponding frontal face-shapes by using linear regression. In [9], the authors employ a 2D face-shape model, based on sparse face-meshes composed of sixty-two facial points extracted from training images, to identify the parameters of the model responsible for the head-pose. Given a test image and its associated mesh, the pose parameters are then set to typical values of frontal faces, thus, obtaining a virtual frontal mesh, i.e., the locations of the facial points in the frontal pose. However, these methods deal only with expressionless facial images, hence, avoiding the elaborate task of decoupling the head-pose and facial expression.

Another approach to facial-point-based head-pose normalization that does not rely on face-shape models has been proposed by Rudovic et al. [14, 13]. In this approach, the authors proposed a GP regression model for learning complex and high-dimensional mappings between 2D facial points, extracted from expressive facial images in non-frontal poses, and corresponding facial points in the frontal pose images. However, the standard GP regression used in this approach accounts for correlations only *between* the input and the output dimensions, but not *within* the input and the output dimensions of the model. This may result in an anatomically impossible facial configuration formed by the predictions of this model. The method we propose in this paper addresses the aforementioned limitations of standard GP regression, and, to the best of our knowledge, is the first one capable of dealing with expressive faces in various poses, and in the presence of noise and outliers in test data.

The outline of the proposed method is shown in Fig.1. The method maps the locations of 39 facial points extracted from a facial image in a non-frontal pose to the corresponding facial points in the frontal pose. To learn the target mappings so that only anatomically feasible solutions arise from the model, we propose a novel GP regression model, which attains structured prediction based on a face-shape model. Here, *structured prediction* refers to exploiting both dependencies between inputs and outputs, and internal dependencies within inputs only and within outputs only. This is in contrast to standard GP regression, which learns only input–output dependencies. We learn the internal dependencies within inputs (outputs) by means of a face-shape model - 2D-PDM - and include them in the model through a face shape prior, defined for both the input and the output of the model. Therefore, the newly proposed GP regression, coined as *Shape-constrained GP (SC-GP) regression*, prevents anatomically impossible solutions (that may be caused by noisy and outlying data) through constraints enforced by the face-shape prior.

The rest of the paper is organized as follows. Section 2 summarizes the basics of GP regression. In Section 3 we describe the utilized deformable face-shape model. Section 4 presents the proposed SC-GP regression. Note that the proposed structured prediction in a probabilistic regression framework could also be attempted by alternative GP-based models including Twin GP [4], Dependent Output GP [5], and Sparse Convolved GP [1] for multi-output regression. However, the latter two models are too complex for the target problem given the high output dimensionality of the model (39×2 dimensions), and so the only alternative left is Twin GP regression. Therefore, in Section 5 we present results for the head-pose normalization attained by the proposed SC-GP regression, and by standard GP regression, Twin GP regression and 3D-face-shape-based methods. Section 6 concludes the paper.

2. Gaussian Process Regression

Gaussian Process (GP) regression has become popular because it is simple to implement, flexible (i.e., it can learn complex mappings through a simple parameterization), and fully probabilistic - which enables us to compare different

models based on their likelihood, and to obtain uncertainty in their predictions [12, 16].

Given a training set $\mathcal{D} = \{(X_i, Y_i)|i = 1, ..., N\}$[1], composed of multi-dimensional inputs $X_i = [x_i^1 ... x_i^D] \in \Re^{N\times D}$ and outputs $Y_i = [y_i^1 ... y_i^D] \in \Re^{N\times D}$, where D is the dimension, the goal of regression is to learn the mapping between the inputs and outputs. From the weight-space view of GP [12] regression, this mapping is described as

$$Y = \mathbf{1}_{(N\times 1)}\, \mu_Y + w\phi(X) + \varepsilon, \tag{1}$$

where μ_Y is the mean of the training outputs, $\phi(X)$ is the projection of the inputs X to the feature space of basis functions $\phi(\cdot)$, w are the weights, and ε is noise on the outputs. The prior distribution over w and ε is Gaussian, and it is given by

$$p(w) = G(w|0, \alpha I), \quad p(\varepsilon) = G(\varepsilon|0, \beta I), \tag{2}$$

where the parameters α and β represent the variance of the distribution. Accordingly, the marginal distribution of the training outputs Y is given by

$$p(Y) = G(Y|\mu_Y, K), \tag{3}$$

where K is $N \times N$ covariance matrix with entries

$$K_{ij} = \alpha\phi(X_i)^T\phi(X_j) + \beta\delta_{ij}, \tag{4}$$

and δ_{ij} is the Kronecker delta function, which is one iff $i = j$, and zero otherwise. By applying the kernel trick [2], the occurrences of the inner product $\langle \phi(X_i)^T, \phi(X_j)\rangle$ in K_{ij} can be replaced with a kernel function evaluated at the input pairs (X_i, X_j). We use the Gaussian kernel, $k(X_i, X_j) = exp(-\theta \|X_i - X_j\|^2)$, since it constraints nearby inputs to have highly correlated outputs [4]. The parameters (θ, β) of the covariance function in Eq.(4) are estimated by minimizing the negative log-likelihood

$$\mathcal{L} = \frac{D}{2}\ln|K| + \frac{1}{2}tr(K^{-1}YY^T) + const., \tag{5}$$

using the conjugate gradient algorithm [12].

Inference of a test input X_* is done by computing the parameters of the predictive distribution $p(Y_*|Y)$, where the estimated output Y_* is conditioned on the observed outputs Y. Since this distribution is also Gaussian with mean and covariance function given by

$$m(X_*) = \mu_Y + K_*^T K^{-1}(Y - \mathbf{1}_{(N\times 1)}\, \mu_Y), \tag{6}$$

$$\sigma^2(X_*) = K_{**} - K_*^T K^{-1} K_*, \tag{7}$$

where $K_* = K(X, X_*)$ denotes $N \times 1$ matrix of the covariances evaluated on the training data X and test datum X_*. The entries of $K = K(X, X)$ and $K_{**} = K(X_*, X_*)$ are computed in the same way, and the the mean $m(X_*)$ is used as the target output Y_*.

[1]The inputs are linearly rescaled to have zero mean and unit variance on the training set.

3. Deformable Face-shape Model

In this section, we describe a deformable face-shape model used later in Sec.4.1 to learn the face-shape prior. In general, deformable shape models offer a unique and powerful approach to face analysis that is capable of accommodating different sources of variation (e.g. facial expressions, identity, etc.). To learn a deformable face-shape model from training data X (defined in Sec.2), we follow standard shape representation where the vector X_i is approximated as

$$X_i \approx \mu_X + c_{X_i} B_X^T, \tag{8}$$

where μ_X is the mean face computed from training data X, $c_{X_i} = [c_{X_i}^{(1)}, ..., c_{X_i}^{(k)}] \in \Re^{1\times k}$ are the shape parameters corresponding to k ($k < D$) deformable modes, which are stored in $B_X = [b_X^{(1)}, ..., b_X^{(k)}] \in \Re^{D\times k}$. Thus, the vector X_i can be reconstructed using the deformable shape model with the parameters $S_{X_i} = (\mu_X, B_X, c_{X_i})$. These parameters are learned by means of standard Principal Component Analysis (PCA) [2]. Although the deformable shape model with parameters obtained in this way is relatively robust to noise in test data, it is highly sensitive to outliers in test data (e.g., caused by occlusions, erroneous hand labelling of the facial points, and/or inaccurate automatic facial point localization), due to the least-squares formulation of standard PCA. Hence, in the presence of outliers, we employ Robust PCA [7] proposed by De La Torre and Black. In both standard PCA and Robust PCA, mean μ_X and deformable modes B_X are learned off-line[2] from training data X, while the shape parameters c_{X_*} for a test input X_* are obtained on-line, as explained in [2, 7].

4. Shape-constrained GP (SC-GP) Regression

In this section, we describe the proposed SC-GP regression for facial-point-based head-pose normalization. We illustrate the method on the task of learning the mapping between 2D locations of the facial points (see Fig.1) in one of non-frontal poses (e.g., $(0°, -45°)$), denoted as $X \in \Re^{N\times D}$, and the corresponding points in the frontal pose $(0°, 0°)$, denoted as $Y \in \Re^{N\times D}$. In what follows, we first describe the face-shape prior that is placed on both the input and the output of GP regression in order to constrain the estimated output to anatomically possible facial configurations. We then describe the optimization procedures for training and inference in the proposed method.

4.1. Face-shape Prior

In standard GP regression, described in Sec.2, the output dimensions are assumed to be independent. How-

[2]We select the number of the deformable modes such that RMSE$< \eta$, where RMSE (as defined in Sec.5) is computed between the original and PCA-reconstructed training data, and η is set to one pixel measured in the registered image plane.

ever, in most cases, modeling complex internal dependences within multi-dimensional inputs and outputs improves regression[4]. Specifically in our case, modeling the spatial correlations between the positions of the facial points is essential as their constellation should satisfy certain anthropomorphic constraints in order to represent an anatomically feasible facial configuration. Fortunately, these geometric constraints can be automatically learned by the deformable face-shape model explained in Sec.3. Hence, we incorporate the information about face-shape into standard GP regression using the deformable face-shape model. Formally, this is attained by defining a face-shape prior that acts as the weight prior in Eq.(2), which is a zero mean Gaussian with the covariance α given by

$$\alpha(S_i, S_j) = p(S_{X_i}, S_{X_j})p(S_{Y_i}, S_{Y_j}), \tag{9}$$

where the covariance $\alpha(S_i, S_j)$ is data-driven and it measures similarity of the input-output data pairs (i, j), based on the corresponding facial shapes (S_i, S_j), defined in Sec.3. The goal of the face-shape prior is to enforce the training data pairs (i, j), with similar input face-shapes, (S_{X_i}, S_{X_j}) (learned from (X_i, X_j)), to have similar output face-shapes, (S_{Y_i}, S_{Y_j}) (learned from (Y_i, Y_j)). The joint probabilities in the above equation are defined as

$$\begin{aligned} &p(S_{X_i}, S_{X_j}) \\ &= \exp(-\tfrac{1}{2}(c_{X_i} - c_{X_j})T_X^{-1}(c_{X_i} - c_{X_j})^T), \end{aligned} \tag{10}$$

$$\begin{aligned} &p(S_{Y_i}, S_{Y_j}) \\ &= \exp(-\tfrac{1}{2}(c_{Y_i} - c_{Y_j})T_Y^{-1}(c_{Y_i} - c_{Y_j})^T), \end{aligned} \tag{11}$$

and they quantify the distance between two face-shapes. The entries of the scaling matrices $T_X = diag(\tau_X^1, ..., \tau_X^k)$ and $T_Y = diag(\tau_Y^1, ..., \tau_Y^k)$ are learned as explained in Sec.4.2.

SC-GP regression is then defined by placing the above-defined face-shape prior on the covariance function from Eq.(4). The entries of the covariance function in SC-GP regression then become

$$K_{ij} = \alpha(S_i, S_j)(k(X_i, X_j) + \beta\delta_{ij}), \tag{12}$$

where the kernel function $k(\cdot, \cdot)$ and noise β are already defined in Sec.2. This covariance function ensures that, in the case of test data corrupted by high levels of noise (or outliers), the model relies more on the face-shape prior than on the noisy inputs.

4.2. SC-GP: Training

The training in SC-GP regression is carried out as follows. First, we learn the deformable models independently for the inputs X and outputs Y, and the number of deformable modes E_b is selected so that $\max(\|X - X^{pca}\|, \|Y - Y^{pca}\|) < \eta$, where η is set manually[3]. Second, we learn E_b pairs of deformable models, $S_X^k = \{B_X^k, C_X^k, \mu_X^k\}$ and $S_Y^k = \{B_Y^k, C_Y^k, \mu_Y^k\}$ with the number of deformable modes from 1 to E_b. Third, we use these deformable models along with the training data X and Y to learn E_b SC-GP regression models (each time with different pair of deformable models) by minimizing $\mathcal{L}^k(X, Y, C_X^k, C_Y^k, hp^k)$ (see Eq.5) w.r.t. hyper-parameters $hp^k = (\tau_X^k, \tau_Y^k, \theta, \beta)$ defined in Sec.4.1. Finally, to select the optimal number of the deformable modes k_{opt} to be used during inference of the test data, we compare the learned SC-GP regression models, and select k_{opt}= $\min_{k=1..E_b}(\mathcal{L}^k(X, Y, C_X^k, C_Y^k, hp^k)$ corresponding to the most likely SC-GP regression model. The parameters used for the inference mode are stored as $\{S_X^{k_{min}}, S_Y^{k_{min}}, X, Y, hp^{k_{min}}\}$.

4.3. SC-GP: Inference

Inference in standard GP regression model is straightforward: we compute a weight matrix based on a test input X_*, and use it to estimate the output Y_*, which is obtained as a weighted combination of the training output data Y. However, in SC-GP regression, to estimate the output Y_* (defined in Eq.(6)), that is given by

$$Y_* = \mu_Y + K_*(c_{Y_*}, c_{X_*})^T K^{-1} Y, \tag{13}$$

we need both shape parameters c_{X_*} and c_{Y_*}. While the parameters c_{X_*} are obtained from the test input X_* (as explained in Sec.3), c_{Y_*} are unknown since they depend on the output to be estimated, Y_*. Thus, this is a chicken-and-egg problem: to estimate the output Y_* we need the shape parameters c_{Y_*}, and the other way around. We approach this problem by using the following two strategies:

1) **Direct approach:** we learn a set of linear ridge regressors (LRRs) independently trained for each output dimension. For a test input X_*, we first obtain an estimate of the output, $\widehat{Y}_*$, using LRRs. Then, we estimate the shape parameters $\widehat{c}_{Y_*}$ from the initial guess $\widehat{Y}_*$. The final output Y_* is obtained by evaluating Eq.(13) using X_*, c_{X_*} and $\widehat{c}_{Y_*}$.

2) **Iterative approach:** we first apply LRRs to obtain an initial estimate of the output, $\widehat{Y}_*^0$, and the corresponding shape parameters, $\widehat{c}_{Y_*}^0$. We then reconstruct the initial estimate $\widehat{Y}_*^0$, either by standard PCA or Robust PCA, using the shape parameters $\widehat{c}_{Y_*}^0$, in order to obtain Y_*^{pca}. Next, we search for the shape parameters $\widehat{c}_{Y_*}$ so that the output of SC-GP regression given by Eq.(13), and Y_*^{pca} are as close as possible. In this way, we iteratively examine the best candidate output shapes until convergence and, based on that, we update the SC-GP-predicted facial landmarks in the frontal

[3]Although we select the same number of deformable modes for both the input and output deformable models, in general, a different number of modes can be chosen for these two.

view. To this end, we minimize L_2-norm of the cost function

$$L(c_{Y_*}) = K_*(c_{Y_*})^T K^{-1} Y^T - Y_*^{pca}, \quad (14)$$

w.r.t. the unknown shape parameters

$$c_{Y_*} = \arg\min_{c_{Y_*}^{(i)}} \left\| L(c_{Y_*}^{(i)}) \right\|. \quad (15)$$

This non-linear optimization problem is solved using a second order quasi-Newton optimizer with cubic polynomial line search for optimal step size selection, which uses the gradient of the objective function at $c_{Y_*}^{(i)}$, given by

$$\frac{\partial L(c_{Y_*})}{\partial c_{Y_*}^{(i)}} = \frac{L(c_{Y_*})}{\|L(c_{Y_*})\|} \cdot \left(\frac{\partial K_*(c_{Y_*})}{\partial c_{Y_*}^{(i)}} \right)^T K^{-1} Y^T, \quad (16)$$

where the gradient of the test covariance K_* (defined in Sec.2) at $c_{Y_*}^{(i)}$ is given by

$$\frac{\partial K_*(c_{Y_*})}{\partial c_{Y_*}^{(i)}} = \begin{bmatrix} -\frac{1}{\tau_2^{(i)}}(c_{Y_*}^{(i)} - c_{Y_1}^{(i)}) K_{*1} \\ \vdots \\ -\frac{1}{\tau_2^{(i)}}(c_{Y_*}^{(i)} - c_{Y_N}^{(i)}) K_{*N}, \end{bmatrix} \quad (17)$$

and the initial shape parameters $c_{Y_*}^0$ are set to $\widehat{c}_{Y_*}^0$.

5. Experiments

We validated our approach using synthetic data from the BU-3D Facial Expression (BU3DFE) database [17], and two real-image datasets: the CMU Pose, Illumination and Expression Database (MultiPie) [10] and a multi-pose facial expression database recorded in our lab. All data were first registered per pose by applying an affine transformation learned using the three facial points: the nasal spine point and the inner corners of the eyes which were chosen since they are stable facial points, and are not affected by facial expressions. The registered data was then used to learn the regression models independently for each target pair of poses (a non-frontal and the frontal pose). The accuracy of the pose-normalization was measured using the Root-Mean-Square-Error (RMSE) defined as $\sqrt{\frac{1}{d}\|\Delta p\|^2}$, where Δp is the difference between the predicted pixels' positions of the facial landmarks in the frontal pose and the ground truth (the manually annotated landmarks in the frontal pose images). If not stated otherwise, in each of the presented experiments, the datasets were partitioned in a person-independent manner and used in a 5-fold cross validation procedure.

In experiments that follow, we compared the performance of the proposed SC-GP regression to that obtained by standard GP regression[4] and the state-of-the-art Twin GP regression[5] [4]. We also compared SC-GP method to: (1) the nonlinear 3D Point Distribution Model (3D-PDM)[19], and (2) the Candide model, being the ASM part of the online AAM [8] we used to localize the facial landmarks in real images (see Sec.5.2).

Table 1. RMSE (per expression) of head pose normalization attained by GP, direct SC-GP, iterative SC-GP, Twin GP, and 3D-PDM, trained on the **BU3DFE** data in 12 training poses and tested on the **BU3DFE** data in 70 test poses

Method	Expression							Av.
	Neutral	Surprise	Disgust	Joy	Anger	Fear	Sadness	
GP	1.45	2.51	2.31	2.31	2.12	1.97	1.73	2.04
SC-GP (dir.)	1.38	2.22	2.03	1.82	1.64	1.52	1.61	1.75
SC-GP (iter.)	1.32	**2.03**	**1.86**	1.71	1.48	1.40	1.62	1.64
Twin-GP	1.13	**2.15**	**1.97**	1.57	1.27	1.20	1.40	1.52
3D-PDM	2.12	2.83	2.58	2.55	2.25	2.39	2.07	2.40

5.1. Performance on Synthetic Data

In the experiments with the synthetic data, we rendered 2D multi-view expressive images from the BU3FE dataset at pan angles from $0°$ to $-45°$, and tilt angles from $0°$ to $30°$ with the step of $5°$, which resulted in 70 poses in total. Only 12 poses (i.e., the poses sampled with the step of $15°$), were used to train the models, while all the 70 poses were used for testing. The data in each pose included expressive images of 50 subjects (54% female), showing six basic facial expressions (joy, sadness, anger, surprise, fear, and disgust, sampled at four different levels of intensity) plus neutral, which resulted in 1250 images per pose. For each of those images, we extracted 2D locations of 39 facial landmarks illustrated in Fig.1, based on the 3D facial points provided by the database creators. These 2D facial points were further used as the features in our experiments. For SC-GP regression, we used the first 16 principal components (deformable modes) computed using the standard PCA, as described in Sec.4.2. In the case of 3D-PDM, we selected 18 deformable modes. Note that the evaluated regression models were trained independently for each pair of a non-frontal pose and the frontal pose, and tested by using the model in the closest training pose (if test data did not belong to any of the training poses). As can be seen from Table 1, in the case of noise-free data, SC-GP regression performed better than GP regression and 3D-PDM. Twin GP outperformed SC-GP regression on average, although iter-

[4]We chose GP regression for head pose normalization as the baseline method for comparison since it has been shown to outperform other regression models, including Linear regression and Support Vector regression, and to perform comparably to Relevance Vector regression on the target task [14, 15]. Also, the recently proposed Coupled GP regression outperforms the baseline GP but it does so only in the case of missing data [13], which is beyond the scope of this paper. Hence, no comparison to this model has been included.

[5]The implementation of Twin GP regression has been obtained from the authors' webpage: http://ttic.uchicago.edu/ blf0218/software/TGP.htm

Table 2. **Influence of different levels of noise and outliers** on head-pose normalization (in terms of RMSE) attained by GP, direct SC-GP, iterative SC-GP, Twin GP, and 3D-PDM. The regression models were trained on the **BU3DFE** noise-free data in 12 training poses, and tested on the **BU3DFE** data in 70 test poses corrupted by different levels of uniformly distributed noise UNIF $\sim [-\alpha, \alpha]$, with $\alpha = 0..5$ pixels ($\alpha = 5$ is 10% of interoccular distance for the registered average frontal-pose face in the **BU3DFE** dataset), and by different levels of bias, $\beta = 0..25$ pixels, added to the locations of 3–5 randomly selected facial points

Method	α						β					
	0	1	2	3	4	5	0	5	10	15	20	25
GP	2.04	2.10	2.21	2.31	2.72	3.10	2.09	2.36	2.88	3.56	3.90	3.99
SC-GP (dir.)	1.75	1.80	1.92	2.04	2.22	2.35	1.84	2.12	2.41	2.63	2.81	2.95
SC-GP (iter.)	1.64	1.70	**1.82**	1.92	2.05	2.14	1.73	**1.99**	2.35	2.51	2.68	2.79
Twin-GP	1.52	1.53	**1.85**	2.22	2.60	3.09	1.55	**2.11**	2.70	3.20	3.38	3.62
3D-PDM	2.40	2.70	2.81	2.93	2.97	2.99	2.45	2.61	2.78	2.95	3.20	3.37

ative SC-GP outperformed Twin-GP in the cases of facial expressions of Surprise and Disgust (the errors shown in bold). These two expressions are more challenging to normalize than the other expressions due to the high variation of their facial landmarks. In addition, note that in real-world applications, where automatic point detectors and trackers are applied, noise-free data (corresponding to highly accurate point detection and tracking) cannot be attained by the existing methods (e.g.,[11]).

In order to investigate the robustness of SC-GP regression to noise and outliers in test data, we ran two sets of experiments: on noisy data and outlying data. First we evaluated the performance of the models in the presence of noise in the BU3DFE test data corrupted by adding five levels of uniformly distributed noise. As can be seen from Table 2 (RMSE values for α), SC-GP regression clearly outperforms GP regression and 3D-PDM. Although Twin GP performs better than SC-GP in the case of low noise levels ($\alpha < 2$), the results clearly suggest that SC-GP is more robust to higher levels of noise. The robustness of SC-GP regression to noise comes from the shape regularization attained by the face shape prior (as explained in Sec.4.1), resulting in effective recovery of shape details from very noisy observations. In addition, iterative SC-GP regression outperformed direct SC-GP regression by iteratively examining the best candidate shapes and updating the predicted facial landmarks in the frontal pose.

In realistic applications, the input data may contain undesirable artifacts due to occlusions, changes in illumination, or inaccurate face/facial point detection/tracking, resulting in *outlying* observations deviating markedly from majority of the training samples. Hence, we evaluated the performance of SC-GP regression in the presence of outliers using the BU3DFE test data corrupted by adding different levels of bias to locations of 3–5 randomly selected facial points. In this experiment, within SC-GP regression, we used the first 20 deformable modes computed by Robust PCA, as described in Sec.4.2. To attain a fair basis for comparison with alternative methods, the test data was preprocessed first by Robust PCA, in order to reduce the effect of the outliers. Otherwise, the results of the directly applied alternative methods to the corrupted data were very poor.

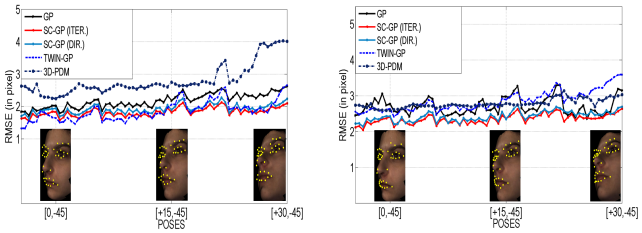

Figure 2. RMSE (per pose) obtained by the regression models trained on the **BU3DFE** noise-free data in 12 training poses, and tested on the **BU3DFE** data in 70 test poses corrupted by the noise level $\alpha = 2$ (*left*) and bias level $\beta = 10$ (*right*)

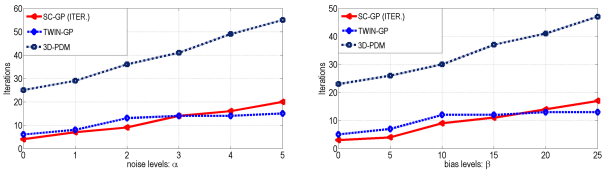

Figure 3. **Number of iterations** required for iterative SC-GP, Twin-GP and 3D-PDM, trained on the **BU3DFE** noise-free data in 12 training poses, to converge when tested on the **BU3DFE** data in 70 test poses, corrupted by noise levels α (*left*) and bias levels β (*right*)

For example, in the case of $\beta = 10$, the average RMSE is 3.9 for GP, 3.6 for Twin GP, and 4.2 for 3D-PDM, where for iterative and direct SC-GP is 2.4 and 2.5, respectively. However, as can be seen from Table 2 (RMSE values for β), even in the case of the preprocessed outlying data, standard GP regression, Twin GP regression, and 3D-PDM were all outperformed by SC-GP (applied directly to unprocessed data). In this case, the robustness of SC-GP regression to outlying data comes from the fact that the model relied almost solely on the face shape prior. As before, and for the same reasons, iterative SC-GP outperformed its direct counterpart.

Fig.2 shows the generalization ability of the models across 70 poses (only 12 of which were used for training) in the presence of the intermediate noise level ($\alpha = 2$), and outliers ($\beta = 10$). In the case of noise (Fig.2(left)), all the models except 3D-PDM were able to generalize well across the poses. More specifically, SC-GP and Twin-GP regression perform comparably (with SC-GP outperform-

Table 3. RMSE (per expression) of head pose normalization attained by GP, direct SC-GP, iterative SC-GP, and Twin GP, trained/tested using the **MultiPie** data in the four discrete poses

Method	Expression				Av.
	Neutral	Surprise	Disgust	Joy	
GP	1.84	2.47	2.25	2.23	2.20
SC-GP (dir.)	1.41	1.91	1.74	1.57	1.67
SC-GP (iter.)	1.45	1.80	1.66	1.52	**1.61**
Twin-GP	1.52	2.08	1.92	1.76	1.82

ing Twin-GP in poses further away from frontal), while both outperform the head pose normalization attained by standard GP regression. The poor performance of 3D-PDM in poses towards $(+30, -45)$ is due to the occlusion of certain facial points in 2D face-images that occurs in those poses. However, in Fig.2 (right) we see that, in the case of outliers, the performance of 3D-PDM improves because of the use of Robust PCA, as the preprocessing step. We also see that Twin-GP is very sensitive to outliers in the test inputs. This is due to the fact that KL distance minimized in Twin-GP regression is not robust to the inputs with non-Gaussian distribution. Fig.3 shows performance of SC-GP regression, Twin-GP regression, and 3D-PDM in terms of the number of iterations required by these models to converge when tested on noisy/outlying data. Both iterative SC-GP and Twin-GP regression converged considerably faster than 3D-PDM. Specifically, these models converged, on average, in 9.6, 10.3 and 34 iterations in the case of noise, and 11.6, 9.5 and 39 iterations in the case of outliers, respectively.

5.2. Performance on Real Data

In experiments on real image data, we used the MultiPie dataset: images of 50 subjects (22% female) displaying 4 facial expressions (neutral, disgust, surprise, and joy) captured at 4 pan angles ($0°$,$-15°$,$-30°$ and $-45°$), resulting in 200 images per pose. These images were annotated in terms of 39 hand-labeled landmark points. For SC-GP regression, we used the first 7 deformable modes computed by standard PCA. As can be seen from Table 3, in the case of real image data, both SC-GP and Twin-GP regression clearly improve standard GP regression, while SC-GP outperforms Twin GP. Although the facial landmarks were manually annotated, this does not guarantee a 'perfect' annotation, especially in cases were some of the points are not clearly visible in the image due to the head pose. Hence, errors in annotation must be expected, introducing additional non-linearities in the mapping to be learned, which, evidently, cannot be well handled by standard GP nor by Twin GP regression.

We also performed experiments on real image sequences recorded in our lab. This dataset contains image sequences of 3 persons (33% female) displaying six basic facial expressions and neutral while rotating their head in front of the camera (starting from frontal pose), comprising the poses at pan angles from $0°$ to $-45°$. The locations of the 39 facial landmark points were obtained by applying the online AAM [8] (see Fig.4). The Candide model from this AAM was also used to attain head pose normalization by rotating it to the frontal pose in order to obtain the 2D-image coordinates. The regression models were trained/tested as explained above, and, for each sequence, the Candide model was manually fitted in the first frame, and the corresponding 2D points obtained from this model were used as the ground truth when computing the RMSE. Table 4 summarizes the average RMSEs per expression computed for all the image sequences. As can be seen, SC-GP outperforms Twin GP regression for all facial expressions.

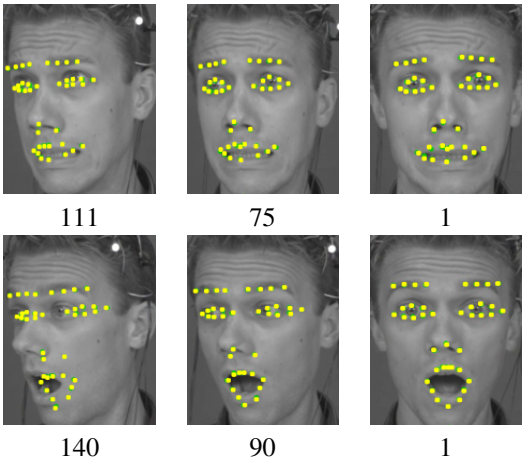

Figure 4. **Our Database:** Sample images (with the automatically tracked facial points) from two image sequences depicting Fear (*top*) and Surprise (*bottom*) while the head-pose is changing from $(0°, 0°)$ to $(0°, -45°)$. The corresponding frame numbers are given below each image.

Fig.5 summarizes the computed RMSEs per frame for two image sequences shown in Fig.4. Note that in poses being far from frontal, the utilized tracker [8] estimates the locations of the facial points less accurately than in near-frontal poses, which resulted in GP and Twin-GP being outperformed by SC-GP regression. The superior performance of SC-GP is due to the use of deformable face-shape model learned per training pair of poses, which enables handling the tracking errors. Note also that all the regression models achieved better results compared to those obtained by the Candide model. This is mainly due to the fact that, during the tracking, the 2D points in non-frontal poses were difficult to align accurately to the corresponding 3D face shape (i.e., the Candide model).

5.3. SC-GP vs. Twin GP

We briefly comment here on the difference between SC-GP and Twin GP. Twin GP is devised to capture correlations within the output dimensions by minimizing the KL divergence between two GP, modeled as normal distributions over training and test data [4]. Yet, when noise and/or

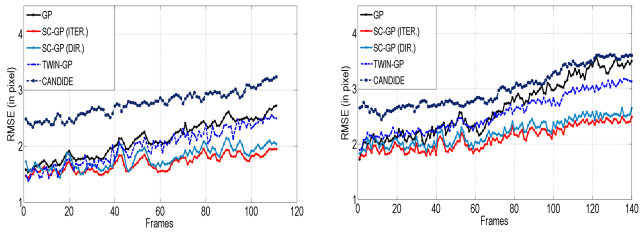

Figure 5. RMSE (per frame) of head pose normalization for two image sequences (Fear – *left* and Surprise – *right*), shown in Fig.4, attained by GP, direct SC-GP, iterative SC-GP, Twin-GP, and the Candide model utilized by the tracker [8]. The models were trained using data of the other two subjects from **Our Database**

outliers exist in test data (e.g., due to the use of automatic facial point localization), the Gaussian assumption in KL divergence is violated by the data. Hence, for the problem at hand, minimizing KL divergence, as done in Twin-GP, cannot guarantee that the output, i.e., the pose-normalized facial points, will form an anatomically feasible facial configuration. By contrast, in the proposed SC-GP, we implicitly impose the constraints, which are determined by the deformable face-shape model, on the input and the output of GP, which, in turn, enforces an anatomically feasible facial configurations in the output of the model. This is the main reason why our model performs better than Twin GP, in the case of noise and outliers in test data, and in the case of real-image data, where automatic facial point localization is applied.

To ensure that improvement in the RMSE of SC-GP(iter.) is statistically significant compared to that of Twin GP, we run t-test ($p = 0.05\%$) for the results obtained in Table 2, 3 and 4, i.e., when SC-GP is expected to outperform Twin-GP, as explained above. For the results in Table 2, the difference in the RMSE of the proposed SC-GP (iter.) and Twin GP is statistically significant in the case of alpha>=3, and theta>=10. For the results in Table 3, we run the significance tests per expression, and altogether. All the differences are statistically significant except for the Neutral, which is also expected. Similarly, for the results in Table 4, we find that all the differences are statistically significant.

Table 4. RMSE (per expression) of head pose normalization attained by GP, direct SC-GP, iterative SC-GP, Twin GP, and Candide model, trained/tested in the subject independent manner using the data from **Our Database**

Method	Expression							Av.
	Neutral	Surprise	Disgust	Joy	Anger	Fear	Sadness	
GP	2.38	4.44	3.80	3.48	3.23	2.85	3.26	3.35
SC-GP (dir.)	2.04	3.22	3.11	2.46	2.34	2.15	2.47	2.60
SC-GP (iter.)	2.00	3.07	2.83	2.59	2.24	2.12	2.49	**2.48**
Twin-GP	2.40	3.47	3.26	3.07	2.59	2.70	2.91	2.91
Candide	3.28	4.36	4.00	4.18	3.45	3.52	3.38	3.74

6. Conclusion

In this paper, we have proposed a novel GP regression model for facial-point-based head-pose normalization. We have shown that the proposed method generalizes well across poses, handles successfully noise and outliers in test data, and is computationally efficient. The proposed model outperforms standard GP regression, 3D-PDM and AAM on the target task. In addition, the proposed model performs comparable to or better than the state-of-the-art Twin GP regression.

Acknowledgments

This work has been funded by the European Research Council under the ERC Starting Grant agreement no. ERC-2007-StG-203143 (MAHNOB).

References

[1] M. Alvarez and N. Lawrence. Sparse convolved multiple output gaussian processes. In *NIPS 2008*, pages 57–64. 2

[2] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2007. 3

[3] V. Blanz, P. Grother, J. Phillips, and T. Vetter. Face recognition based on frontal views generated from non-frontal images. In *CVPR 2005*, pages 454–461. 2

[4] L. Bo and C. Sminchisescu. Twin gaussian processes for structured prediction. *Int. J. Comput. Vision*, 87(1-2):28–52, 2010. 2, 3, 4, 5, 7

[5] P. Boyle and M. Frean. Dependent gaussian processes. In *NIPS 2005*, pages 217–224. 2

[6] T. Cootes, G. Wheeler, K. Walker, and C. Taylor. View-based active appearance models. *Image and Vision Computing*, 20(9-10):657 – 664, 2002. 2

[7] F. De La Torre and M. J. Black. A framework for robust subspace learning. *Int. J. Comput. Vision*, 54(1-3):117–142, 2003. 3

[8] F. Dornaika and J. Orozco. Real time 3d face and facial feature tracking. *J. Real-Time Image Processing*, 2(1):35–44, 2007. 2, 5, 7, 8

[9] D. Gonzalez-Jimenez and J. Alba-Castro. Symmetry-aided frontal view synthesis for pose-robust face recognition. In *ICASSP 2007*, pages II–237 –II–240, 2007. 2

[10] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 28(5):807 – 813, 2010. 5

[11] S. Milborrow and F. Nicolls. Locating facial features with an extended active shape model. In *ECCV*, 2008. 6

[12] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005. 3

[13] O. Rudovic, I. Patras, and M. Pantic. Coupled gaussian process regression for pose-invariant facial expression recognition. In *ECCV 2010*, pages 350–363. 2, 5

[14] O. Rudovic, I. Patras, and M. Pantic. Facial expression invariant head pose normalization using gaussian process regression. In *CVPR-W 2010*, pages 28–33. 2, 5

[15] O. Rudovic, I. Patras, and M. Pantic. Regression-based multi-view facial expression recognition. In *ICPR 2010*, pages 4121–4124. 5

[16] R. Urtasun and T. Darrell. Sparse probabilistic regression for activity-independent human pose inference. In *CVPR*, pages 1–8, 2008. 3

[17] J. Wang, L. Yin, X. Wei, and Y. Sun. 3d facial expression recognition based on primitive surface feature distribution. In *CVPR 2006*, pages 1399–1406. 5

[18] T. Wang and J. Lien. Facial expression recognition system based on rigid and non-rigid motion separation and 3d pose estimation. *Pattern Recognition*, 42(5):962 – 977, 2009. 2

[19] Z. Zhu and Q. Ji. Robust real-time face pose and facial expression recovery. In *CVPR 2006*, pages 681–688. 2, 5