# A Linear Approach of 3D Face Shape and Texture Recovery using a 3D Morphable Model

Oswald Aldrian

Department of Computer Science
The University of York

William A. P. Smith

**Abstract**

In this paper, we present a robust and efficient method to statistically recover the full 3D shape and texture of faces from single 2D images. We separate shape and texture recovery into two linear problems. For shape recovery, we learn empirically the generalization error of a 3D morphable model using out-of-sample data. We use this to predict the 2D variance associated with a sparse set of 2D feature points. This knowledge is incorporated into a parameter-free probabilistic framework which allows 3D shape recovery of a face in an arbitrary pose in a single step. Under the assumption of diffuse-only reflectance, we also show how photometric invariants can be used to recover texture parameters in an illumination insensitive manner. We present empirical results with comparison to the state-of-the-art analysis-by-synthesis methods and show an application of our approach to adjusting the pose of subjects in oil paintings.

## 1 Introduction

3D Morphable Models (3DMM) have been used for modelling face appearance for over a decade [1]. They represent the state-of-the-art in pose and illumination insensitive face recognition [9] due to their explicit image formation model. This means the statistical model need only capture intrinsic face variation caused by changes in identity, leaving extrinsic factors to be estimated as part of the fitting process. Despite this, there have been relatively few developments since 3DMMs were first introduced. There are two obvious developments required to improve their performance for face analysis tasks. The first is to improve the generalisation ability of the models to overcome problems of model dominance. The second is to develop new methods for fitting the model to images which improve efficiency and robustness.

In this paper we focus on a new, highly efficient means to accurately fit a 3DMM in a pose and illumination insensitive manner. Most previous work in this area has taken the form of iterative optimisation in an analysis-by-synthesis framework. The idea here is to iteratively refine estimates of parameters describing shape, texture, illumination, pose and camera properties such that the error between predicted and observed appearance is minimised. This is a highly complex and expensive optimisation, littered with local minima. Results are also



BMVC 2010 doi:10.5244/C.24.75

dependent on a regularisation parameter which trades off implausibility against a low data error. If the model prior is allowed to dominate, the results appear face-like but loose their distinctiveness. Data dominance on the other hand, results in over-fitting and faces that appear unrealistic and caricature like.

There have been some attempts to address these problems. At the expense of simplifying the reflectance assumptions by using a Lambertian model, Zhang and Samaras [11] showed that a morphable model could be fitted under unknown and arbitrarily complex illumination conditions using a spherical harmonic basis. On the other hand, both Romdhani and Vetter [8] and Moghaddam *et al*. [6] focus on improving the accuracy and efficiency of the fitting process respectively. In both cases, they avoid the problems of local minima by using features derived from the input images rather than the intensity data itself. Romdhani and Vetter [8] used edges and specular highlights to obtain a smooth cost function, while Moghaddam *et al*. [6] used silhouettes computed from a large number of input images. Knothe *et al*. [5] have begun to consider the problem of model dominance and used local feature analysis to locally improve the fit of the model to a set of sparse feature points.

The closest work in spirit to what we present here is that of Romdhani *et al*. [10]. They present linear solutions to computing an incremental update to the shape and texture parameters given dense measurements of residual errors provided by optical flow. However, their iterative approach requires nonlinear optimisation of pose parameters and illumination terms. In this paper, we propose a shape estimation method which incorporates an empirically measured model of variance into a linear objective function. By doing so, we do not need a weight factor that trades off between the model and the data. We recover texture using a linear error based on a photometric invariant which is unaffected by illumination conditions. At the expense of assuming diffuse-only reflectance and that the location of sparse feature points are known, we are able to accurately fit a 3DMM at greatly reduced computational expense whilst closely competing with the accuracy of much more sophisticated methods. We present the first quantitative comparative evaluation of 3DMM fitting algorithms.

## 2 3D Morphable Models

A 3D morphable model is constructed from $m$ face meshes which are in dense correspondence. Each mesh consists of $p$ vertices and is written as a vector $\mathbf{v} = [x_1\ y_1\ z_1\ \ldots\ x_p\ y_p\ z_p]^T \in \mathbb{R}^n$, where $n = 3p$. Applying principal components analysis to the data matrix formed by stacking the $m$ meshes provides us with $m-1$ eigenvectors $\mathbf{S}_i$, their corresponding variances $\sigma_{s,i}^2$ and the mean shape $\bar{\mathbf{v}}$. An equivalent model is constructed for surface texture (or more precisely, diffuse albedo). Any face can be approximated as a linear combination of the modes of variation:

$$\mathbf{v} = \bar{\mathbf{v}} + \sum_{i=1}^{m-1} a_i \mathbf{S}_i, \qquad \mathbf{u} = \bar{\mathbf{u}} + \sum_{i=1}^{m-1} b_i \mathbf{T}_i, \tag{1}$$

where $\mathbf{a} = [a_1\ \ldots\ a_{m-1}]^T$ and $\mathbf{b} = [b_1\ \ldots\ b_{m-1}]^T$ are vectors of parameters. For convenience, we also define the variance-normalised shape parameter vector as:
$\mathbf{c}_s = [a_1/\sigma_{s,1}\ \ldots\ a_{m-1}/\sigma_{s,m-1}]^T$.

# 3 Shape Parameter Estimation

We present a novel algorithm for shape parameter estimation under unknown pose given the 2D coordinates of a sparse set of $N \ll p$ feature points. In order to obtain a linear solution, we decompose the problem into two steps which can be iterated and interleaved: 1. estimation of a camera projection matrix using known 3D-2D correspondences, and 2. estimation of 3D shape parameters using a known camera projection matrix. We initialise by using the mean shape to compute an initial estimate of the camera projection matrix, $\mathbf{C} \in \mathbb{R}^{3\times4}$. With this to hand, shape parameters can be recovered using only matrix multiplications. By using the recovered shape to re-estimate the camera matrix, we can iterate the process which typically converges in only 3 iterations. A good solution is still achieved with only one pass of our method.

## 3.1 Estimating the Camera Projection Matrix

We represent the 2D locations of feature points in the image, $\mathbf{x}_i \in \mathbb{R}^3$, and corresponding 3D locations of the feature points within the model, $\mathbf{X}_i \in \mathbb{R}^4$, as homogeneous coordinates. To estimate the camera projection matrix, we require normalised versions: $\tilde{\mathbf{x}}_i = \mathbf{T}\mathbf{x}_i$ and $\tilde{\mathbf{X}}_i = \mathbf{U}\mathbf{X}_i$, where $\mathbf{T} \in \mathbb{R}^{3\times4}$ and $\mathbf{U} \in \mathbb{R}^{4\times4}$ are similarity transforms which translate the centroid of the image/model points to the origin and scale them such that the RMS distance from the origin is $\sqrt{2}$ for the image points and $\sqrt{3}$ for the model points.

We assume an affine camera and compute the normalised projection matrix, $\tilde{\mathbf{C}} \in \mathbb{R}^{3\times4}$, using the *Gold Standard Algorithm* [4]. Given $N \geq 4$ model to image point correspondences $\mathbf{X}_i \leftrightarrow \mathbf{x}_i$, we determine the maximum likelihood estimate of $\tilde{\mathbf{C}}$ which minimises: $\sum_i \|\tilde{\mathbf{x}}_i - \tilde{\mathbf{C}}\tilde{\mathbf{X}}_i\|^2$, subject to the affine constraint $\tilde{\mathbf{C}}_3 = [0\,0\,0\,1]$. Each point correspondence contributes to the following $2N \times 8$ system of equations:

$$\begin{bmatrix} \tilde{\mathbf{X}}_1^T & \mathbf{0}^T \\ \mathbf{0}^T & \tilde{\mathbf{X}}_1^T \\ \vdots & \vdots \\ \tilde{\mathbf{X}}_N^T & \mathbf{0}^T \\ \mathbf{0}^T & \tilde{\mathbf{X}}_N^T \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{C}}_1^T \\ \tilde{\mathbf{C}}_2^T \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{x}}_{1,1} \\ \tilde{\mathbf{x}}_{1,2} \\ \vdots \\ \tilde{\mathbf{x}}_{N,1} \\ \tilde{\mathbf{x}}_{N,2} \end{bmatrix}. \tag{2}$$

We solve this system using least squares and obtain the camera matrix by performing the following de-normalization step: $\mathbf{C} = \mathbf{T}^{-1}\tilde{\mathbf{C}}\mathbf{U}$.

## 3.2 A Probabilistic Approach

We recover the 3D shape parameters using a probabilistic approach which follows that of Blanz *et al*. [2]. However, our derivation is more complex as we allow different 2D variances, $\sigma_{2D,i}^2$, for each feature point. We discuss how these variances are computed in the next section. Our aim is to find the most likely shape vector $\mathbf{c}_s$ given an observation of $N$ 2D features points in homogeneous coordinates: $\mathbf{y} = [x_1\ y_1\ 1\ \ldots\ x_N\ y_N\ 1]^T$ and taking into account the model prior. From Bayes' rule we can state: $P(\mathbf{c}_s|\mathbf{y}) = \nu \cdot P(\mathbf{y}|\mathbf{c}_s) \cdot p(\mathbf{c}_s)$, where $\nu = (\int P(\mathbf{y}|\acute{\mathbf{c}}_s) \cdot p(\acute{\mathbf{c}}_s) d\acute{\mathbf{c}}_s)^{-1}$ is a constant factor. The coefficients are normally distributed with zero mean and unit variance, i.e. $\mathbf{c}_s \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$, so the probability of observing a given $\mathbf{c}_s$ is: $p(\mathbf{c}_s) = \nu_c \cdot e^{-\frac{1}{2}\|c_s\|^2}$, where $\nu_c = (2\pi)^{-m'/2}$. The probability of observing the data $\mathbf{y}$

for a given $\mathbf{c}_s$ is simply:

$$P(\mathbf{y}|\mathbf{c}_s) = \prod_{i=1}^{3N} \nu_N \cdot e^{-\frac{1}{2\sigma_{2D,i}^2}[y_{model2D,i}-y_i]^2}. \tag{3}$$

Here, $y_{model2D,i}$ are the homogeneous coordinates of the 3D feature points projected to 2D, defined as follows. We construct the matrix $\hat{\mathbf{S}} \in \mathbb{R}^{3N\times m-1}$ by subselecting the rows of the eigenvector matrix $\mathbf{S}$ associated with the $N$ feature points. We further modify this matrix by inserting a row of zeros after every third row of $\mathbf{S}$, giving the matrix $\hat{\mathbf{S}}_h \in \mathbb{R}^{4N\times m-1}$. In other words, the directions in 3D along which a vertex is perturbed according to an eigenvector are written in homogeneous coordinates. We now form the block diagonal matrix $\mathbf{P} \in \mathbb{R}^{3N\times 4N}$ in which the camera matrix is placed on the diagonal:

$$\mathbf{P} = \begin{bmatrix} \mathbf{C} & & \\ & \ddots & \\ & & \mathbf{C} \end{bmatrix} \tag{4}$$

Finally, we can define the 2D points obtained by projecting the 3D model points given by $\mathbf{c}_s$ to 2D: $y_{model2D,i} = \mathbf{P}_i \cdot (\hat{\mathbf{S}}_h diag(\sigma_s^2)\mathbf{c}_s + \bar{\mathbf{v}})$, where $\mathbf{P}_i$ is the $i$th row of $\mathbf{P}$.

Substituting into Bayes' rules, we arrive at our conditional probability:

$$P(\mathbf{c_s}|\mathbf{y}) = \nu \cdot \nu_N^l \cdot \nu_c \cdot e^{-\sum_{i=1}^{3N} \frac{[y_{model2D,i}-y_i]^2}{2\sigma_{2D,i}^2}} \cdot e^{-\frac{1}{2}\|\mathbf{c}_s\|^2}, \tag{5}$$

which can be maximised by minimising the exponent:

$$E = -2 \cdot logP(\mathbf{c}_s|\mathbf{y}) = \sum_{i=1}^{3N} \frac{[y_{model2D,i}-y_i]^2}{\sigma_{2D,i}^2} + \|\mathbf{c}_s\|^2 + \text{const.} \tag{6}$$

This is very similar to a *Tikhonov Regularization* using a Gaussian kernel. However, as opposed to Blanz *et al*. [2], we measure the variances for each individual feature point and therefore do not need a weight factor which relates the prior to the data as the relationship between them is determined empirically. $\|\mathbf{c}_s\|^2$ corresponds to a regularisation term which measures the complexity of the functional and penalises 'complicated solutions'. In terms of the statistical model, this means that solutions closer to the mean are preferred.

## 3.3 Modelling Feature Point Variance

We model two sources of variance which can be used to explain the difference between observed and modelled feature point positions in the image. By having an explicit model of this variance, we negate the need for an ad hoc regularisation weight parameter. The first source of variance is the generalisation error of the morphable model. This describes how feature points deviate from their true position in 3D when the optimal model parameters are used to describe a face. The second source of variance is the 2D pixel noise, this is related to the accuracy with which the feature points can be marked up in 2D. The total variance of a feature point is the sum of the 3D variance projected to 2D and the 2D variance.

Given an out-of-sample face mesh $\mathbf{v}_i$ (i.e. a face that was not used to train the statistical model), we project onto the model to obtain the closest (in a least squares sense) possible approximation: $\mathbf{v}_i' = \mathbf{S}\mathbf{S}^T(\mathbf{v}_i - \bar{\mathbf{v}}) + \bar{\mathbf{v}}$. The vector of squared errors is given by: $\mathbf{e}_i = (\mathbf{v}_i - \mathbf{v}_i')^2$.

We define $\hat{\mathbf{e}}_i$ as the vector formed by subselecting the elements of $\mathbf{e}_i$ which correspond to the $N$ sparse feature points. From a sample of $k$ such out-of-sample faces, we can now compute the variance associated the coordinates of the feature points: $\sigma^2_{3D,j} = \frac{1}{k}\sum_{i=1}^{k} \hat{\mathbf{e}}_{i,j}$. This gives us an empirical means to predict how a feature point is likely to vary from its true position due to generalisation errors. The units of $\sigma_{3D,j}$ are mm. In order to predict how this results in variation in the image plane, we must project these variances to 2D, in units of pixels. The 3D variance of the $i$th feature point in homogeneous coordinates is given by: $[\sigma^2_{3D,3i-2}\ \sigma^2_{3D,3i-1}\ \sigma^2_{3D,3i}\ 1]^T$. We define $\mathbf{C}_{-T} \in \mathbb{R}^{3\times 4}$ as the camera projection matrix without translational components. This is required because the variances are with respect to the feature point position and do not need globally translating. Our final 2D variances are given by the sum of the projected 2D variances and a 2D pixel error, $\eta^2$, which models error in feature point markup:

$$\begin{bmatrix} \sigma^2_{2D,3i-2} \\ \sigma^2_{2D,3i-1} \\ \sigma^2_{2D,3i} \end{bmatrix} = \mathbf{C}_{-T} \begin{bmatrix} \sigma^2_{3D,3i-2} \\ \sigma^2_{3D,3i-1} \\ \sigma^2_{3D,3i} \\ 1 \end{bmatrix} + \begin{bmatrix} \eta^2 \\ \eta^2 \\ 0 \end{bmatrix}. \tag{7}$$

We use a value of $\eta^2 = 3$ in our experiments.

## 3.4 Minimizing the Error Functional

In this section we describe how the error functional in (6) can be minimized in a single step. To do so, we differentiate the functional, set to zero and solve for $\mathbf{c}_s$. Therefore the constant factor in the equation is not relevant and the functional takes the following form:

$$E = \sum_{i=1}^{3N} \frac{[y_{model2D,i} - y_i]^2}{\sigma^2_{2D,i}} + \|\mathbf{c}_s\|^2. \tag{8}$$

Substituting the statistical model into (8), applying the second binominal theorem and rewriting yields:

$$E = \sum_{i=1}^{3N} \frac{[\mathbf{P}_i \cdot \hat{\mathbf{S}}_h\mathbf{a} + \mathbf{P}_i \cdot \bar{\mathbf{v}}]^2 - 2[\mathbf{P}_i \cdot \hat{\mathbf{S}}_h\mathbf{a} + \mathbf{P}_i \cdot \bar{\mathbf{v}})]y_i + y_i^2}{\sigma^2_{2D,i}} + \|\mathbf{c}_s\|^2. \tag{9}$$

For clarity, we introduce two constants: $\mathbf{R}_i = \mathbf{P}_i\hat{\mathbf{S}}_h$ and $k_i = 2\mathbf{P}_i \cdot \bar{\mathbf{v}}$. Expanding according to the first binominal theorem we obtain:

$$E = \sum_{i=1}^{3N} \frac{(\mathbf{R}_i\mathbf{a})^2 + k_n(\mathbf{R}_i\mathbf{a}) + (\mathbf{P}_i \cdot \bar{\mathbf{v}})^2 - 2y_i\mathbf{R}_i\mathbf{a} + k_iy_i + y_i^2}{\sigma^2_{2D,i}} + \|\mathbf{c}_s\|^2. \tag{10}$$

We would like to minimise the error so we differentiate with respect to $\mathbf{a}$ and set the derivative to zero:

$$0 = \nabla E = \sum_{i=1}^{3N} \frac{2\mathbf{R}_i^T\mathbf{R}_i\mathbf{a} + k_i\mathbf{R}_i^T - 2y_i\mathbf{R}_i^T}{\sigma^2_{2D,i}} + 2\mathbf{c}_s \tag{11}$$

Since we wish to solve the system of equations for the normally distributed coefficients $\mathbf{c}_s$ instead of $\mathbf{a}$, we multiply the vector $\mathbf{R}_n$ by the shape eigenvalues, $\mathbf{Q}_i = \mathbf{R}_i diag(\sigma^2_{s,i})$, and obtain:

$$\sum_{i=1}^{3N} \frac{2\mathbf{Q}_i^T\mathbf{Q}_i\mathbf{c}_s}{\sigma^2_{2D,i}} + 2\mathbf{c}_s = \sum_{i=1}^{3N} \frac{2y_i\mathbf{Q}_i^T}{\sigma^2_{2D,i}} - \sum_{i=1}^{3N} \frac{k_i\mathbf{Q}_i^T}{\sigma^2_{2D,i}}. \tag{12}$$

For simplicity we set:

$$\mathbf{T}_1 = \sum_{i=1}^{3N} \frac{2\mathbf{Q}_i^T \mathbf{Q}_i}{\sigma_{2D,i}^2} \quad \text{and} \quad \mathbf{T}_2 = \sum_{i=1}^{3N} \frac{2y_i \mathbf{Q}_i^T}{\sigma_{2D,i}^2} - \sum_{i=1}^{3N} \frac{k_i \mathbf{Q}_i^T}{\sigma_{2D,i}^2} \tag{13}$$

and obtain the following equation: $\mathbf{T}_1\mathbf{c}_s + 2\mathbf{c}_s = \mathbf{T}_2$. This can be solved by applying a *Cholesky Decomposition* to $\mathbf{T}_1$ and decomposing the result further with a *Singular Value Decomposition*:

$$\begin{aligned}
\mathbf{M}^T\mathbf{M}\mathbf{c}_s + 2\mathbf{c}_s &= \mathbf{T}_2, \quad \text{where:} \quad \mathbf{T}_1 = \mathbf{M}^T\mathbf{M} && (14)\\
\mathbf{V}\mathbf{W}^2\mathbf{V}^T\mathbf{c}_s + 2\mathbf{c}_s &= \mathbf{T}_2, \quad \text{where:} \quad \mathbf{M} = \mathbf{U}\mathbf{W}\mathbf{V}^T && (15)\\
diag(w_i+2)\mathbf{V}^T\mathbf{c}_s &= \mathbf{V}^T\mathbf{T}_2 && (16)\\
\mathbf{c}_s &= [diag(w_i+2)\mathbf{V}^T]^{-1}\mathbf{V}^T\mathbf{T}_2. && (17)
\end{aligned}$$

Hence, using only a sequence of matrix multiplications, we are able to recover the maximum likelihood estimate of $\mathbf{c}_s$ given the location of projected 2D feature points and the projection matrix, allowing for the variances of each feature point to differ.

# 4 Linear Texture Recovery from Photometric Invariants

Our statistical surface texture model captures variations in diffuse albedo. This forms one parameter of a number of possible parametric reflectance models (e.g. Phong) which in turn determines the appearance of a face [1]. This is the approach used in analysis-by-synthesis model fitting. We take a different approach. By making some assumptions about the surface reflectance and illumination, we are able to arrive at a photometric invariant which can be measured directly from the image and used to fit the texture model in an illumination-insensitive manner. Moreover, the resulting solution is linear in terms of the observed image intensities and can therefore be executed efficiently.

We make the assumption that surface reflectance is diffuse only and that illumination is provided by any combination of directional and ambient white sources:

$$I_{\{r,g,b\}} = \rho_{\{r,g,b\}} \int_{\Omega_\mathbf{N}} V_\omega L(\omega)(\mathbf{N}\cdot\omega)d\omega, \tag{18}$$

where $\Omega_\mathbf{N}$ is the hemisphere about the surface normal $\mathbf{N}$, $L(\omega)$ is the incident radiance from direction $\omega$ and $V_\omega$ is the visibility function, equal to 1 if direction $\omega$ is unoccluded, 0 otherwise. The important observation is that ratios between pairs of colour channels are functions of only the ratio of albedos, hence we can relate ratios of texture model values directly to image intensity ratios:

$$\frac{\mathbf{T}_{r(i)}\mathbf{b} + \bar{u}_{r(i)}}{\mathbf{T}_{b(i)}\mathbf{b} + \bar{u}_{b(i)}} = \frac{I_{r(i)}}{I_{b(i)}} \quad \text{and} \quad \frac{\mathbf{T}_{g(i)}\mathbf{b} + \bar{u}_{g(i)}}{\mathbf{T}_{b(i)}\mathbf{b} + \bar{u}_{b(i)}} = \frac{I_{g(i)}}{I_{b(i)}}. \tag{19}$$

where $\mathbf{T}_{r(i)}$ and $\bar{u}_{r(i)}$ represent the eigenvector and mean value for a corresponding observation $I_{r(i)}$ (in this case for the red channel). Image intensities are measured by sampling the image at the position of all visible (i.e. unoccluded) vertices in the face mesh. See Figure (1) for an example.

Figure 1: Mapping a 2D image onto the projected 3D shape model. The shape is reconstructed using 59 feature points.

Equation (19) can be rewritten as follows:

$$(I_{b(i)}\mathbf{T}_{x(i)} - I_{x(i)}\mathbf{T}_{b(i)})\mathbf{b} = I_{x(i)}\bar{u}_{b(i)} - I_{b(i)}\bar{u}_{x(i)}, \tag{20}$$

where the index $x$ is substituted for $r$ or $g$ respectively. This gives us a linear system of equations of the following form:

$$\underbrace{\begin{pmatrix} I_{b(1)}\mathbf{T}_{r(1)} - I_{r(1)}\mathbf{T}_{b(1)} \\ I_{b(1)}\mathbf{T}_{g(1)} - I_{g(1)}\mathbf{T}_{b(1)} \\ \vdots \\ I_{b(k)}\mathbf{T}_{r(k)} - I_{r(k)}\mathbf{T}_{b(k)} \\ I_{b(k)}\mathbf{T}_{g(k)} - I_{g(k)}\mathbf{T}_{b(k)} \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} b_1 \\ \\ \vdots \\ \\ b_{m-1} \end{pmatrix}}_{\mathbf{b}} = \underbrace{\begin{pmatrix} I_{r(1)}\bar{u}_{b(1)} - I_{b(1)}\bar{u}_{r(1)} \\ I_{g(1)}\bar{u}_{b(1)} - I_{b(1)}\bar{u}_{g(1)} \\ \vdots \\ I_{r(k)}\bar{u}_{b(k)} - I_{b(k)}\bar{u}_{r(k)} \\ I_{g(k)}\bar{u}_{b(k)} - I_{b(k)}\bar{u}_{g(k)} \end{pmatrix}}_{\mathbf{h}}, \tag{21}$$

with two equations per observed pixel value and can be solved using least-squares: $\mathbf{b} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{h}$. A minimum of $k = m/2$ image-model correspondences is necessary to solve the system for $m$ model parameters. In practice, many thousands of visible pixels are used.

# 5 Experiments and Results

Our experiments are based on the *Basel Face Model* [7]. The model is accompanied by 10 out of sample faces. Each face is rendered in 9 poses and 3 lighting conditions per pose giving 270 renderings in total. We use a subset of the Farkas feature points [3] to reconstruct face shape. Depending on the pose, different feature points are visible. In a frontal view, we use up to 66 feature points and in the close to profile views (-70,70) as less as 37 feature points are visible. We use the 60 most significant eigenmodes to reconstruct the face. For most of the renderings more modes are possible. But to be consistent, we choose the number according to the least number of visible feature points. We compare our results against the state-of-the-art analysis-by-synthesis result. To make the results comparable, we take only the first 60 shape coefficients into account. We quantify the reconstruction error in terms of the mean Euclidean error over all vertices in the mesh. We conducted our experiments on the full set of 270 renderings. Figure (2) shows the mean error for the individual poses for all 10 faces and 3 lighting conditions.

Comparing each of the individual renderings results in 191 lower errors for the analysis-by-synthesis approach [7] and 79 lower errors for our method. Figure (3) shows a snippet

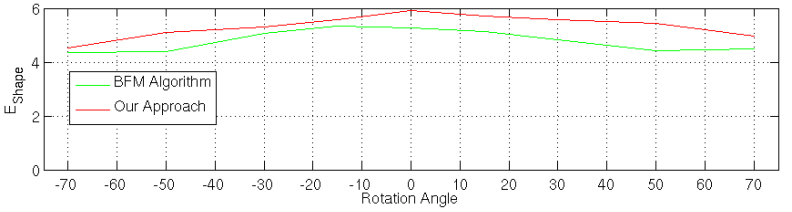


Figure 2: Reconstruction errors for all faces and lighting conditions plotted over the rotation angles $(-70, -50, -30, -15, 0, 15, 30, 50, 70)$.

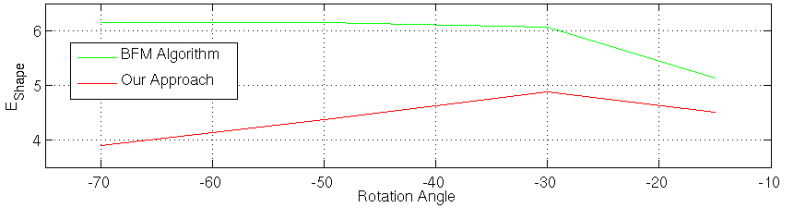


Figure 3: Reconstruction errors for faces No. 4 in lighting condition 3 plotted over the rotation angles $(-70, -50, -30, -15)$.

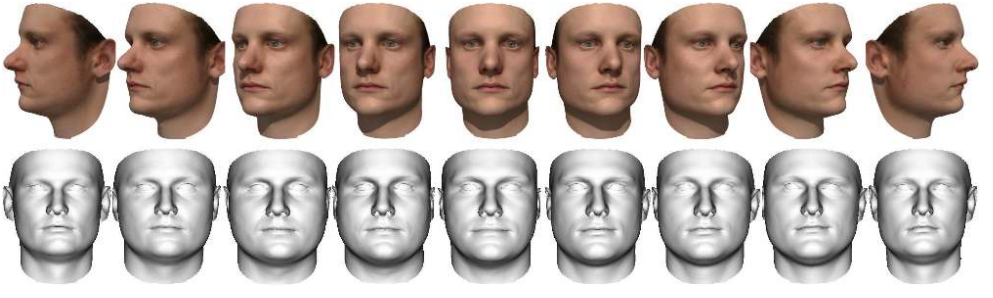

Figure 4: Top row shows 2D renderings of face No. 4 in different rotation angles $(-70° : 70°)$. Second row shows the corresponding 3D shape reconstructions.

of face No. 4 in the pose angles $(-70° : -15°)$ and lighting condition 3. Figure (4) shows shape reconstructions of a sample face in 9 different pose angles.

As with the shape, we conducted texture reconstruction experiments for the full set of 270 2D renderings. We measure the error to the ground truth texture in terms of root-mean-square error. Based on that measure, the mean texture reconstruction error for our method is 0.18 compared to 0.09 for the analysis-by-synthesis approach [7]. However, our use of photometric invariants renders our result more stable. Figure (5) shows the difference between the highest error and the lowest texture error for all faces in the 9 pose angles. This demonstrates the stability of our texture recovery under varying illumination.

In figure (6) we show the shape and texture reconstruction for face No. 7 in frontal pose under the 3 different lighting conditions. Note the presence of cast shadows in the image which do not effect our method. Also, these renderings include specular reflections which are not modelled in our photometric invariant. Finally, in Figure (7) we demonstrate an application of our method to adjusting the pose of a subject in an oil painting.

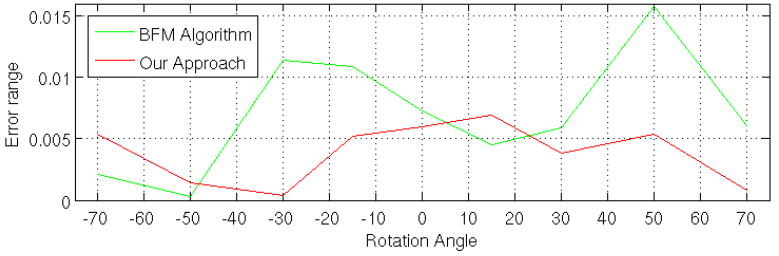


Figure 5: The relative error between the individual faces for different pose angles.

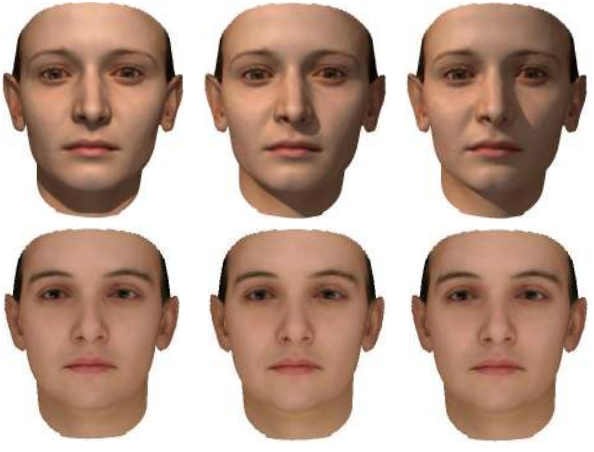

Figure 6: Top row shows renderings of Face No. 7 in frontal pose and 3 different illumination conditions. Second row shows the corresponding shape and texture reconstructions.

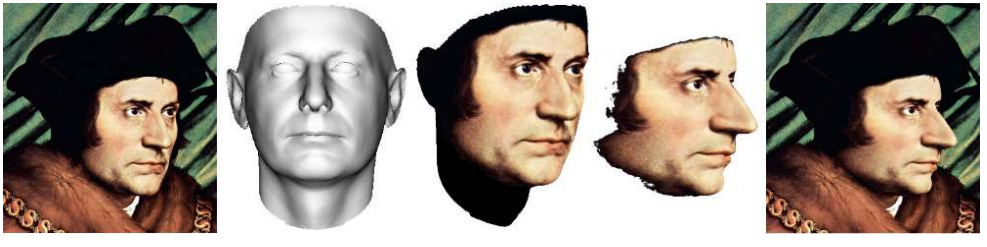

Figure 7: From left to right: subject in oil painting, reconstructed 3D shape in frontal view, projected 3D shape with texture mapped on it, cropped and rotated version, oil painting with adjusted pose.

# 6 Conclusion

We have presented a linear approach to face shape and texture estimation using a morphable model. The accuracy of our approach is comparable to a state-of-the-art analysis-by-synthesis algorithm, yet is orders of magnitude faster (less than a second using unoptimised Matlab code versus several minutes [1]). In addition, our empirical model of generalisation error was learnt using only 10 out-of-sample faces. Increasing this would likely improve

results. Our experiments also showed that the number of feature points alone is not the significant factor. On average, the shape reconstruction error is lower for close to profile views compared to front views, even though nearly half as many feature points are visible. This implies that the pose of a face effects the information content in a feature point observation. Our texture estimation is based on photometric invariants and our experimental results show the output is almost unaffected by changes in illumination. In future work, we will explore using more complex invariants which are also robust to specular reflections.

# References

[1] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *Proc. SIGGRAPH*, pages 187–194, 1999.

[2] V. Blanz, A. Mehl, T. Vetter, and H-P. Seidel. A statistical method for robust 3d surface reconstruction from sparse data. In *Proc. 3DPVT*, pages 293–300, 2004.

[3] L. Farkas. *Anthropometry of the Head and Face*. Raven Press, New York, 1994.

[4] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521623049, 2000.

[5] R. Knothe, S. Romdhani, and T. Vetter. Combining PCA and LFA for surface reconstruction from a sparse set of control points. In *Proc. Int. Conf. on Automatic Face and Gesture Recognition*, pages 637–644, 2006.

[6] B. Moghaddam, J. Lee, H. Pfister, and R. Machiraju. Model–based 3–D face capture with shape–from–silhouettes. In *Proc. IEEE Work. Analysis and Modeling of Faces and Gestures*, pages 20–27, 2003.

[7] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter. A 3D face model for pose and illumination invariant face recognition. In *Proc. IEEE Intl. Conf. on Advanced Video and Signal based Surveillance*, 2009.

[8] S. Romdhani and T. Vetter. Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *Proc. CVPR*, volume 2, pages 986–993, 2005.

[9] S. Romdhani, J. Ho, T. Vetter, and D. J. Kriegman. Face recognition using 3–D models: Pose and illumination. *Proc. of the IEEE*, 94(11):1977–1999, 2006.

[10] Sami Romdhani, Volker Blanz, and Thomas Vetter. Face identification by fitting a 3d morphable model using linear shape and texture error functions. In *in European Conference on Computer Vision*, pages 3–19, 2002.

[11] L. Zhang and D. Samaras. Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(3):351–363, 2006.