3D Probabilistic Feature Point Model for Object Detection and Recognition

Sami Romdhani Thomas Vetter
University of Basel, Computer Science Department,
Bernoullistrasse 16, CH - 4056 Basel, Switzerland

Abstract

This paper presents a novel statistical shape model that can be used to detect and localise feature points of a class of objects in images. The shape model is inspired from the 3D Morphable Model (3DMM) and has the property to be viewpoint invariant. This shape model is used to estimate the probability of the position of a feature point given the position of reference feature points, accounting for the uncertainty of the position of the reference points and of the intrinsic variability of the class of objects. The viewpoint invariant detection algorithm maximises a foreground/background likelihood ratio of the relative position of the feature points, their appearance, scale, orientation and occlusion state. Computational efficiency is obtained by using the Bellman principle and an early rejection rule based on 3D to 2D projection constraints. Evaluations of the detection algorithm on the CMU-PIE face images and on a large set of non-face images show high levels of accuracy (zero false alarms for more than 90% detection rate). As well as locating feature points, the detection algorithm also estimates the pose of the object and a few shape parameters. It is shown that it can be used to initialise a 3DMM fitting algorithm and thus enables a fully automatic viewpoint and lighting invariant image analysis solution.

1. Introduction

3D Morphable Model (3DMM) is a well known method for modelling a class of objects [2]. Using a synthesis algorithm, a 3DMM can be used to produce a photo-realistic image of an object, given its model parameters. Using an analysis-by-synthesis framework, the inverse problem can also be addressed: given an object image, the model parameters that explain this image are estimated. This is an optimisation problem, called *fitting*, usually addressed by gradient descent techniques. It has been shown that 3DMM provides state of the art result in one of the most challenging instance of face recognition: identification in presence of combined pose and illumination variations [3].

Probably one of the major drawback of the 3DMM is its requirement of a careful manual initialisation of the fitting algorithm: Due to the local minima of the optimisation procedure used to estimate the model parameters, it is necessary to start the iterative analysis algorithm close to its optimum. It has been shown in [3] that several face image analysis applications obtain good results when the analysis algorithm is initialised with the position of a few (such as seven) feature points. Typically, these landmark points are marked by a human operator on each face image that is to be analysed. There is no record of a fully automatic analysis algorithm for the 3D Morphable Model. The aim of this paper is to propose a method that addresses this limitation.

Automatically localising feature points in images of a class of objects is a classical detection problem. Detection algorithms may be classified in two groups. The vast majority of the detection algorithms, here denoted as *holistic detection algorithms*, aim at estimating the 2D centre of the object in an image and its scale [19, 22]. Hence, they yield a box around the object of interest. Some of these algorithms are multi-view and they also return a pose estimate [11]. However, this information is not sufficient to initialise the 3DMM analysis algorithm. The second kind of detection algorithms, referred to as *feature point detection algorithms*, which have attracted much attention recently, aims at localising feature points of the object [4, 8, 7].

From a high level perspective both types of detection algorithms work similarly: At a learning stage statistical models of the image variation of the object of interest and of the background are constructed. At detection time, a brute force matching is performed: The solution space is exhaustively sampled and a distance between the statistical model and each of these samples is computed. The object is then said to be detected if the minimum distance is lower than a threshold and the solution sample achieving this minimum provides the localisation of the object.

Holistic detection algorithms represent the object as a cropped image patch. All extrinsic (illumination and pose) and intrinsic (object specific) variations are modeled implicitly. In order to achieve efficient detection algorithms, only a few hundred pixels are usually included in these image patches. Additionally, pixel correspondences between the object image patches are not used. Feature points detection algorithms, on the other hand, represent the object of interest by the appearance of several feature points (with a small image patch around each feature point) and some of them also use their relative 2D position. There is no doubt that a 3DMM models the object of interest more accurately than these detection based models: Because of the

1-4244-1180-7/07/$25.00 ©2007 IEEE

specificity of the 3DMM (which can only generate facial images), we may believe that if detections are performed using the 3DMM as object image model, less false alarms will result.

Clearly, unification of 3DMM detection and fitting is beneficial both for detection and fitting: the detection would be more accurate and the fitting automatic and efficient. Then, the question is how to combine detection and fitting. The goal is to construct a detection algorithm that guarantees that what is detected can then be fitted by the 3DMM. Towards this objective, we propose to detect feature points whose image position are in the span of the shape model of the 3DMM. In fact, this may be a too restrictive requirement, as it would not allow for image noise. Hence, we require that the position of the detected points minimises the Mahalanobis distance to the shape model. Additionally, the appearance of each detected point should match the appearance of the modeled feature point. To achieve these goals, we use a probabilistic detection algorithm similar to the one of Fergus *et al.* [9]. This algorithm combines the Maximum Likelihood (ML) estimation of Weber *et al.* [23], which allows for occluded feature points, with the conditionally independent constellation model used by Felzenszwalb *et al.* [7] that provides efficient global optimisation (reviewed in Section 3). The major difference between the work of Fergus and the one presented in this paper lies in the shape modelling. Here, we introduce a probabilistic shape model (Section 2) for feature points that is based on a 3DMM and is invariant to rigid 3D transformation and flexible deformation of the object. An algorithm using it for the detection problem is detailed in Section 4. It is then compared to similar methods (such as RANSAC) in Section 5. Detection and identification experiments are reported in Section 6.

2. Probabilistic feature points shape model

The probabilistic shape model, introduced here, aims at estimating the probability, $p(\boldsymbol{X}|Object)$, that a set of N_p 2D feature points, $\boldsymbol{X}$, belong to a class of object of interest. Calculating this probability requires integrating over a set of model parameters, θ. If the image noise is independent for each feature point, the likelihood factors into likelihoods of two subsets of feature points,

$$p(\boldsymbol{X}|Object) = \int p(\boldsymbol{X_r}|\theta)p(\boldsymbol{X_{\bar{r}}}|\theta)p(\theta)d\theta. \quad (1)$$

We call these subsets, the reference feature points, $\boldsymbol{X_r}$, and the non-reference feature points, $\boldsymbol{X_{\bar{r}}}$ ($\boldsymbol{r}$ and $\bar{\boldsymbol{r}}$ are the indices of the reference and non-reference points).

If the likelihood, $p(\boldsymbol{X_r}|\theta)$, is peaky, the integral can be approximated by an evaluation of the likelihood at its maximum value. The model parameter value that achieves the maximum likelihood (ML) is denoted by $\hat{\theta} = \arg\max_\theta p(\boldsymbol{X_r}|\theta)$, and using a non-informative prior, the posterior can be reduced to

$$p(\boldsymbol{X}|Object) \approx p(\boldsymbol{X_r}|\hat{\theta})p(\boldsymbol{X_{\bar{r}}}|\hat{\theta}) = p(\boldsymbol{X}|\hat{\theta}). \quad (2)$$

In this setting, a subset of the feature points are used to estimate the ML of the model parameters and the full set of points are used to calculate the likelihood of their position given this ML estimate.

Using, again, the conditional independence of the feature points given the model parameters, we can factor the feature point likelihood as $p(\boldsymbol{X}|\hat{\theta}) = \prod_i^{N_p} p(\boldsymbol{x_i}|\hat{\theta})$. In the next sections, we use the linear object class framework and derive the ML estimate of the model parameters and the likelihood of the last equation.

2.1. Linear object class

A successful approach to flexible deformation modeling is based on the concept of Linear Object Class [2], which assumes that the flexible 3D deformations of a class of object vary linearly and which uses a multivariate Gaussian to model the prior for the model's coefficients. Then, image plane coordinates are obtained by applying a 3D to 2D projection. One instance of Linear Object Class model applied to human faces is the 3D Morphable Face Model [2]. In this framework, a flexible shape model is constructed by applying Principal Component Analysis to a training set of N_p 3D feature points that have been put into correspondence with a reference object. A single PCA model is built for the 3D coordinates by representing the shape of an object as a $3N_p \times 1$ column vector. This leads to $L < 3N_p$ principal components. If the normality assumption of the model's coefficients is valid, then the conditional independence of the feature points given the model parameters assumption made at Equation (1) is also valid.

The 3D shape of an object is then obtained by a linear combination of principal components. In order to compute a weak perspective projection and a 2D translation of this 3D shape, using a single matrix product, a fourth coordinate is added to the 3D shape. Similarly to homogeneous coordinates, this fourth coordinate is set to one for all vertices, by arranging the mean shape, $\boldsymbol{S^0}$, as a $4 \times N_p$ matrix with all elements of the last row set to one. The principal components, $\boldsymbol{S^j}$, are also arranged as $4 \times N_p$ matrix with all elements of the last row set to zero. Then, the 2D image positions of the feature points, denoted by the $2 \times N_p$ matrix $\boldsymbol{X}$, are obtained by multiplying the 2×4 weak-perspective matrix $\boldsymbol{P}$ with the shape matrices as follows.

$$\boldsymbol{X} = \boldsymbol{P}\boldsymbol{S^0} + \sum_{j=1}^{L} \alpha_j \boldsymbol{P}\boldsymbol{S^j}, \quad (3)$$

$$\text{where} \quad \boldsymbol{P} = \begin{pmatrix} \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \end{pmatrix} \cdot R_\gamma R_\zeta R_\phi & t_{2d} \end{pmatrix}. \quad (4)$$

In this equation, f is the focal length, γ is the image plane rotation angle, ζ is the elevation rotation angle, ϕ is the azimuth rotation angle and t_{2d} is a 2D translation.

An important property of PCA is that, if the principal components are scaled by their standard deviation, the joint PDF of the L parameters is a standard Normal: $p(\boldsymbol{\alpha}) \sim$

$N(\mathbf{0}, \boldsymbol{I}_L)$, where $\boldsymbol{I}_L$ is the $L \times L$ identity matrix. In this framework, the model parameter, θ, is composed of the shape coefficients, α_j, and the projection matrix, $\boldsymbol{P}$.

The strategy to calculate the probability of the feature point given the model of the previous section is then the following: (i) The ML of the projection matrix and a subset of shape parameters, $\hat{\boldsymbol{P}}$ and $\hat{\alpha}_j$, are calculated from the 2D coordinates of reference points. Usually, the reference points are detected in an input image and there is an uncertainty in their position. The uncertainty of these detection is expressed as their covariance matrix, $\boldsymbol{\Sigma}_{\boldsymbol{X}_r^d}$[1]. The uncertainty propagation results in a mean and covariance of the ML of the projection and shape parameters, $\boldsymbol{\mu}_{\hat{\boldsymbol{P}}}$, $\boldsymbol{\Sigma}_{\hat{\boldsymbol{P}}}$, $\boldsymbol{\mu}_{\hat{\boldsymbol{\alpha}}}$ and $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\alpha}}}$. (ii) From the first two moments of the ML parameters, the mean and covariance of the position of the reference and non-reference points is computed. Then, $p(\boldsymbol{x_i}|\hat{\boldsymbol{P}}, \hat{\boldsymbol{\alpha}})$ is calculated using a Gaussian model.

The uncertainty of the joint position of the feature points has then two causes: (i) The maximum likelihood of the model parameters is itself uncertain because it is estimated from a noisy feature point detector. (ii) The limited number of reference points used (which enables an efficient detection) may not be sufficient to fully constraint the intrinsic variability of the object class, and thus not all model parameters may be estimated, which increases the uncertainty of the non-reference feature points.

2.2. Maximum likelihood of the model parameters

In this section, it is shown how to estimate the ML of the model parameters, $\hat{\boldsymbol{\alpha}}$ and $\hat{\boldsymbol{P}}$, that explain the reference points. We assume that the reference points are affected by a Gaussian noise with zero mean and covariance matrix denoted by $\boldsymbol{\Sigma}_{\boldsymbol{X}_r^d}$. The ML is then the solution of

$$\{\hat{\boldsymbol{P}}, \hat{\boldsymbol{\alpha}}\} = \arg\min_{\boldsymbol{P},\boldsymbol{\alpha}} \operatorname{vec}(\boldsymbol{P}\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}} + \sum \alpha_j \boldsymbol{P}\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{j}} - \boldsymbol{X}_{\boldsymbol{r}})^{\mathrm{T}} \boldsymbol{\Sigma}_{\boldsymbol{X}_r^d} \operatorname{vec}(\boldsymbol{P}\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}} + \sum \alpha_j \boldsymbol{P}\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{j}} - \boldsymbol{X}_{\boldsymbol{r}}). \tag{5}$$

It is shown in [17] that there exist several ML estimates of the projection matrix, depending on the number of principal components of the flexible model used to estimate it in Equation (5). It turns out that the quality of these estimation differ and the estimate of the projection matrix that is the least impacted by the detected feature points noise is the one obtained with a *selective* estimate that uses only the mean shape matrix (i.e. using the mean of the prior of the shape parameters, which is null). This can be seen as estimating the position of the feature points using a projection of the mean shape and then estimating the residual using the flexible shape model. Then, the ML estimate of the projection matrix, $\hat{\boldsymbol{P}}$, is obtained as follows.

$$\operatorname{vec}(\hat{\boldsymbol{P}}) = (\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}^+})^{\mathrm{T}} \otimes \boldsymbol{I}_2) \operatorname{vec}(\boldsymbol{X}_{\boldsymbol{r}}). \tag{6}$$

[1] Usually, the detection is independent for each feature point and the covariance matrix is diagonal or block diagonal, however, we treat it as a general matrix for this analytical derivation.

The subscript $\cdot_{\boldsymbol{r}}$ denotes the columns of the matrix that pertain to the reference feature points, the superscript $\cdot^+$ denotes the pseudo-inverse and $\otimes$ denotes the Kronecker product. This equation holds if $N_r \geq 4$. As we will see in the next section, this lower bound on the number of reference points has major implications in terms of computational load of the detection algorithm.

The first moments of the PDF of $\operatorname{vec}(\hat{\boldsymbol{P}})$ may be obtained using the uncertainty propagation law [13]. Hence, the mean and covariance of the ML of the projection parameters (Equation (5)) are

$$\begin{aligned} \operatorname{vec}(\boldsymbol{\mu}_{\hat{\boldsymbol{P}}}) &= (\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}^+})^{\mathrm{T}} \otimes \boldsymbol{I}_2) \operatorname{vec}(\boldsymbol{X}_{\boldsymbol{r}}), \\ \boldsymbol{\Sigma}_{\hat{\boldsymbol{P}}} &= (\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}^+})^{\mathrm{T}} \otimes \boldsymbol{I}_2) \boldsymbol{\Sigma}_{\boldsymbol{X}_r^d} (\boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}^+} \otimes \boldsymbol{I}_2). \end{aligned} \tag{7}$$

As the projection is formed of an isotropic scaling and a rotation matrix, it must agree to the following constraints: The norm of the first three elements of its first two rows must be equal and their dot product, null.

$$\|\hat{\boldsymbol{P}}_{1,1:3}\|^2 - \|\hat{\boldsymbol{P}}_{2,1:3}\|^2 = 0, \quad \text{and} \quad \hat{\boldsymbol{P}}_{1,1:3} \hat{\boldsymbol{P}}_{2,1:3}^{\mathrm{T}} = 0. \tag{8}$$

This can be seen as a prior on the projection parameters. We will see in Section 4 that this provides an early rejection rule that is used to speed up the detection algorithm by several orders of magnitude.

The ML estimate of the shape coefficients is then obtained using the mean of the ML estimate of the projection matrix (according to the uncertainty propagation rule). Denoting by $\boldsymbol{A}$ a matrix whose column j is $\operatorname{vec}(\boldsymbol{\mu}_{\hat{\boldsymbol{P}}} \boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{j}})$ for $j = 1, \ldots, M$, yields

$$\operatorname{vec}(\boldsymbol{X}_{\boldsymbol{r}} - \boldsymbol{\mu}_{\hat{\boldsymbol{P}}} \boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}}) = \boldsymbol{A}\hat{\boldsymbol{\alpha}}, \tag{9}$$

where the number of estimated shape coefficient, $M \leq L$, is such that the system of equations is not under-constrained. This yields the mean and covariance matrix of the shape parameters (with $M \leq 2N_r$)

$$\boldsymbol{\mu}_{\hat{\boldsymbol{\alpha}}} = \boldsymbol{A}^+ \operatorname{vec}(\boldsymbol{X}_{\boldsymbol{r}} - \boldsymbol{\mu}_{\hat{\boldsymbol{P}}} \boldsymbol{S}_{\boldsymbol{r}}^{\boldsymbol{0}}), \qquad \boldsymbol{\Sigma}_{\hat{\boldsymbol{\alpha}}} = \boldsymbol{A}^+ \boldsymbol{\Sigma}_{\boldsymbol{X}_r^d} \boldsymbol{A}^{+^{\mathrm{T}}}. \tag{10}$$

2.3. Probability of a feature point given parameters

We can now proceed to the estimation of $p(\boldsymbol{x_i}|\hat{\theta}) = p(\boldsymbol{x_i}|\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{P}})$. According to our model, and denoting by $\boldsymbol{x_i}$ the i^{th} column vector of the matrix $\boldsymbol{X}$ (i.e. the position of i^{th} feature point), we have

$$\boldsymbol{x_i} = (\boldsymbol{S}_{\boldsymbol{i}}^{\boldsymbol{0}^{\mathrm{T}}} \otimes \boldsymbol{I}_2) \operatorname{vec}(\hat{\boldsymbol{P}}) + \sum_{j=1}^{M} \hat{\alpha}_j (\boldsymbol{S}_{\boldsymbol{i}}^{\boldsymbol{j}^{\mathrm{T}}} \otimes \boldsymbol{I}_2) \operatorname{vec}(\hat{\boldsymbol{P}}) + \sum_{j=M+1}^{L} \alpha_j (\boldsymbol{S}_{\boldsymbol{i}}^{\boldsymbol{j}^{\mathrm{T}}} \otimes \boldsymbol{I}_2) \operatorname{vec}(\hat{\boldsymbol{P}}). \tag{11}$$

The third term of this equation accounts for the variability presents in the class of objects and not explained by the reference feature points. The mean of the prior of the shape model coefficients is null, hence application of the uncertainty propagation law yields the following mean.

$$\boldsymbol{\mu}_{\boldsymbol{x_i}} = (\boldsymbol{S_i^{0}}^{\mathsf{T}} \otimes \boldsymbol{I}_2)\,\mathrm{vec}(\boldsymbol{\mu}_{\hat{\boldsymbol{P}}}) + \sum_{j=1}^{M} \mu_{\hat{\alpha_j}} (\boldsymbol{S_i^{j}}^{\mathsf{T}} \otimes \boldsymbol{I}_2)\,\mathrm{vec}(\boldsymbol{\mu}_{\hat{\boldsymbol{P}}}). \tag{12}$$

Let us denote by $\boldsymbol{J_P}$ the derivative matrix of $\mathrm{vec}(\boldsymbol{x_i})$ with respect to $\mathrm{vec}(\hat{\boldsymbol{P}})$ at the mean of the shape coefficients and by $\boldsymbol{j}_{\alpha_j}$, the derivative vector with respect to the shape coefficient j.

$$\begin{aligned} \boldsymbol{J_P} &\doteq \frac{\partial\,\mathrm{vec}(\boldsymbol{x_i})}{\partial\,\mathrm{vec}(\hat{\boldsymbol{P}})} = (\boldsymbol{S_i^{0}}^{\mathsf{T}} \otimes \boldsymbol{I}_2) + \sum_{j=1}^{M} \mu_{\hat{\alpha_j}} (\boldsymbol{S_i^{j}}^{\mathsf{T}} \otimes \boldsymbol{I}_2), \\ \boldsymbol{j}_{\alpha_j} &\doteq \frac{\partial\,\mathrm{vec}(\boldsymbol{x_i})}{\partial\,\mathrm{vec}(\alpha_j)} = (\boldsymbol{S_i^{j}}^{\mathsf{T}} \otimes \boldsymbol{I}_2)\,\mathrm{vec}(\boldsymbol{\mu}_{\hat{\boldsymbol{P}}}). \end{aligned} \tag{13}$$

According to the prior, the shape coefficients are uncorrelated and have unit standard deviation, $p(\boldsymbol{\alpha}) \sim N(\boldsymbol{0}, \boldsymbol{I})$, hence, the covariance matrix of $\mathrm{vec}(\boldsymbol{x_i})$ is

$$\boldsymbol{\Sigma}_{\boldsymbol{x_i}} = \boldsymbol{J_P} \boldsymbol{\Sigma}_{\hat{\boldsymbol{P}}} \boldsymbol{J_P}^{\mathsf{T}} + \sum_{j=1}^{M} \sum_{k=1}^{M} \boldsymbol{j}_{\alpha_j} \boldsymbol{j}_{\alpha_k}^{\mathsf{T}} \sigma_{jk} + \sum_{j=M+1}^{L} \boldsymbol{j}_{\alpha_j} \boldsymbol{j}_{\alpha_j}^{\mathsf{T}}, \tag{14}$$

where σ_{jk} is the element (j,k) of the ML estimate of the shape coefficient covariance matrix $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\alpha}}}$ (Equation (10)). The first two terms of this covariance matrix are due to the uncertainty of the feature point detection, $\boldsymbol{\Sigma}_{\boldsymbol{X_r^d}}$. If the detection is noiseless, these terms are null. The last term models the part of the flexible deformation of the linear object class that is not explained by the reference points. Using the mean (Equation (12)) and the covariance matrix (Equation (14)), $p(\boldsymbol{x_i}|\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{P}})$ can be computed using a Gaussian model. A Gaussian model is chosen, because, assuming that only the first two moments are known, then the maximally non-informative (max entropy) distribution to describe our state of knowledge is the Gaussian model. The number of estimated shape coefficients, M, must be chosen according to $0 \leq M \leq 2N_r$.

Example: Figure 1 shows the contour containing 99% of the probability of the estimation of the position of the mouth corner given four reference points. A Gaussian noise with a STD of 3 pixels was added to all reference points. Note that the ellipses change as the reference points vary with the pose of the face. The area of the ellipses are respectively 18.7^2, 18.6^2 and 17.9^2 pixel square. (The distance between the eyes on the frontal view is 160 pixel.) As can be seen, the uncertainty area is rather small. This shows that the proposed shape model can not only be used for detection applications but also as a general view invariant probabilistic shape model, which could supersede other shape models such as the Active Shape Model [5].

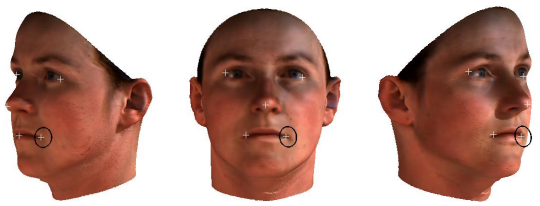


Figure 1. On these synthetic examples, four reference points (marked by a cross), whose locations are perturbed by a Gaussian noise of 3 pixel STD, are used to estimate the PDF of the location of a fifth point (right mouth corner). The ellipses enclosing 99% of the probability are shown.

3. Probabilistic feature points detection

The viewpoint invariant probabilistic shape model introduced in the previous section is applied to feature point detection. We use a similar detection framework to that of Fergus *et al.* [9]. Due to space limitations, we only outline it here. Feature points are searched among key points generated using a generic interest point detector. We chose to use the SIFT detector [14], because it is invariant to image scale and rotation and it is robust to 3D pose and illumination variations, however, any other interest point detector could be used instead.

The task of detecting and localising the object is to find correspondence between the set of N_p model feature points and a subset of the N_k image key points. This set of correspondences is represented by an N_p dimensional hypothesis vector, $\boldsymbol{h}$, whose element i is the index of the key point in correspondence with feature point i. If no such corresponding point exists, then the value of h_i is set to zero, hence, $0 \leq h_i \leq N_k$. The size of the ensemble of values that $\boldsymbol{h}$ can take, is $N_k^{N_p}$.

Let us denote by $\mathcal{K}$ the set of key points extracted from an image. A key point detector, such as the SIFT detector, provides, for each extracted key point, an appearance representation, denoted by $\boldsymbol{a}$, a scale, s, an orientation, o, and its 2D position in the image, $\boldsymbol{x}$. The foreground likelihood can be factorised in the following manner.

$$p(\mathcal{K}|\hat{\theta}) = \sum_{\boldsymbol{h} \in H} \underbrace{p(\boldsymbol{a}|s,o,\boldsymbol{h},\hat{\theta})}_{\text{appearance}} \underbrace{p(s,o|\boldsymbol{h},\hat{\theta})}_{\text{rel. scale and orient.}} \underbrace{p(\boldsymbol{x}|\boldsymbol{h},\hat{\theta})}_{\text{shape}} \underbrace{p(\boldsymbol{h}|\hat{\theta})}_{\text{occlusions}}. \tag{15}$$

We assumed that the feature point appearance does not depend on their position, and that the scale, orientation and occlusion state of a feature point depend on the projection matrix of the object. The model parameter, θ regroup here the shape and projection parameters, $\boldsymbol{\alpha}$ and $\boldsymbol{P}$, and also parameters of the probabilistic models of the appearance, image scale and orientation.

For the background likelihood, there is only one hypothesis, $\boldsymbol{h}_0 = \boldsymbol{0}$ and it is also assumed that the appearance, scale, orientation and localisation of feature points are independent. ML detection is performed by computing the likelihood ratio of the foreground over the background like-

lihood. The object is localised by finding the hypothesis vector, $\boldsymbol{h}$, that maximises the likelihood.

The appearance of each key point is modelled by a Mixture of Gaussians and similarly to Ke *et al.* [12], the representation is based on the image gradient of a patch around the key point. The probability of the scale and orientation of a key point is conditioned on the focal length and on the image plane rotation angle of the object. The occlusion model is conditioned on the azimuth and elevation angles of the object. These angles are derived from the ML of the projection matrix estimated from the reference points (Section 2).

When the logarithm of the likelihood ratio is expanded, it becomes clear that some terms depend on the reference points and the remaining terms constitute a sum over the non-reference points. Each term of this sum depends on the reference points and on a *single* non-reference point. Hence, using Bellman's Principle from Dynamic Programming, minimising the negative log-likelihood can be done by minimising the appropriate term for each non-reference point, h_i and for all combinations of reference points position. As a result, the original problem that was of complexity $N_k^{N_p}$ is now transformed into a problem of complexity $N_k^{N_r+1}$, which makes the detection algorithm much more efficient, as $N_r < N_p$.

4. Feature point detection algorithm

The probabilistic shape model introduced in Section 2 and the detection algorithm outlined in Section 3 are used to simultaneously detect and localise feature points and estimate the projection matrix and some flexible shape parameters. This is achieved by the following algorithm.

1. SIFT point detection: First a SIFT point detector [14] is applied to the input image. This provides a set of key point locations along with their scale and orientation. The number of points, N_k, depends on the size of the input image and usually varies between 300 and 1500. An image patch normalised in scale and orientation around each interest point is then formed whose gradient is coded in a generic PCA with 36 dimensions similarly to PCA-SIFT [12].

2. Appearance model based rejection: Each key point is then rated to each feature point appearance model with the appearance likelihood ratio. If this ratio is low, the key point is rejected from the set of possible feature point i. This is implemented by keeping the N_a key points with the highest appearance ratio for each feature point. Note that, at this stage, a key point may be included in the set of points for several feature points. In our implementation, we use $N_a = 10$.

3. Projection constraint rejection: After the second step, the number of possible combinations of reference points is $N_a^{N_r}$ (for $R = 4$ and $N_a = 10$, there are 10,000 combinations). Each combination gives rise to an hypothetical projection matrix using Equation (7). Many of these projection matrices are not valid, as they do not agree to the constraints of Equation (8). The combinations of feature points leading to invalid projection matrices are rejected. It should be noted that this step is computationally cheap: a projection matrix is obtained by a matrix-vector product and the projection constraints are computed by three dot-products. In practice, it turns out that an average of only 15 combinations of reference points, denoted by H_R, are left after this step (this average is computed on the face images of the validation set detailed in Section 6). Hence, *the speed-up provided by this stage is of three orders of magnitude*. On non-face images, this average is even lower as four random background feature points seldom agree to the projection constraints and from 10,000 combinations on average, 0.6 combinations remain. The projection parameters of the valid projection matrices are extracted, as they are required to compute the relative scale and orientation likelihood and the occlusion likelihood. Then the log likelihood ratios for all valid reference point combinations are computed.

4. Maximum Likelihood estimate: To find the ML estimate, the hypothesis vector, $\boldsymbol{h}$, that minimises the log likelihood ratio is searched among the H_R reference point combinations that agree to the projection constraint and the N_a points for each non-reference points. If the minimum likelihood ratio is higher than a threshold, no object is detected in the input image.

5. Parameters refinement using all visible feature points: Similarly to a RANSAC algorithm [10], if a face has been detected (at step 4), the ML estimate of the projection matrix and of the shape coefficients can be refined. To this end, Equation (5) is maximised with a Gauss-Newton algorithm that estimates jointly the projection matrix and the shape parameters. This is done using all visible feature points of the optimal configuration found at step 4. Alternatively, if the positions of the feature points are also susceptible of improvement, Equation (15) is maximised, instead. In the experiments reported in Section 6, Equation (5) was used.

For $N_p = 10$, $N_a = 10$, $N_r = 4$ and $M = 0$, a Matlab implementation of step 2 of this algorithm runs in 130ms on a 3GHz Pentium IV. Steps 3 and 4 require 28ms. As this paper is not about efficient SIFT point detection, this timing assumes that the SIFT points have been detected and does not include the timing of step 1. This algorithms allows for occlusion of non-reference feature points but not for occlusion of reference feature points. To allow correct detection even when one or many reference points are occluded, we apply steps 3 and 4 of the algorithm to all combinations of N_r reference points among the N_p feature points. The detection is then the minimum of the likelihood ratio over all combinations. For $N_p = 10$ and $N_r = 4$, there are 210 such combinations, making the detection algorithm run in 4s. To improve efficiency, one could stop the algorithm as soon as a combination of feature points is found that yields a likelihood ratio lower than a threshold.

5. Discussions

The shape model and detection algorithm presented here shares some features with the following algorithms.

Local photometry and global geometry, a.k.a. Pictorial structures: Detecting objects by combining information of a series of image patch (treated independently) and of their position in the image is naturally not new. Recently, Fergus *et al.* [8] extended the "soft detection" strategy of [4, 23] (that sought the arrangement of feature points position that jointly maximises a shape likelihood ratio (global geometry) and the responses of the image feature points detectors (local photometry)) using an interest operator to detect image patches that provides their scale. The detection cost function used in the present paper is essentially the same as in [8] (with the exception of the orientation of the image patch that is also used here and the fact that here, the shape model is viewpoint invariant). This line of research addressed the problem of the energy function derivation but no efficient and principled optimisation method was proposed. This limitation was then addressed by the Pictorial Structure work of Felzenszwalb and Huttenlocher [7] that used the conditional independence assumption of the feature points positions to take advantage of the Bellman Principle to efficiently find the global maximum of the likelihood function. Certainly, one of the nicest feature of this algorithm is that real-time detection is obtained even though no hard threshold is set on the image patch detectors prior to joint shape and appearance likelihood maximisation and as a result, the likelihood is maximised over the full solution space (which is not the case in the algorithm presented here due to the appearance based rejection of step 2.). Fergus *et al.* [9] then used this efficient optimisation algorithm to maximise the likelihood ratio of [8]. In [7] and [9], a single feature point is used as reference. Hence, the geometric model is only translation and scale invariant. Recently, Crandall *et al.* [6] proposed the same detection algorithm as the Pictorial Structure work [7] but with a shape model that uses several reference feature points. Surprisingly, this was not done to allow for greater projection invariance (scale, rotation in the image plane, nor out of the image plane), but rather to provide more constraints on the flexible shape model. The authors conclude that no significant detection accuracy is obtained by using more than one reference point. (However, the computational load rises exponentially.) The main difference between these algorithms and the one proposed here is the shape model: The endeavour of the aforementioned algorithms is to train the model in an unsupervised fashion. This is attractive when the objective is to detect and recognise any class of objects from a set of unlabelled training images. Here, we are interested in detecting and localising a specific class of object (human face) for which we have extensive prior knowledge which we want to use to improve detection and localisation accuracy. This enables us to obtain a weak perspective invariant shape model and a detection likelihood ratio that models occlusions using the out of the image plane rotation angle.

RANSAC: The algorithm presented in this paper also share similarities with RANSAC [10] type algorithms and more specifically with MLESAC [21]. MLESAC is casted in the multiple view geometry framework and aims to estimate the relation between consecutive views of a video sequence in terms of a fundamental matrix. Having found putative correspondences between two images, the point is to find the fundamental matrix that maximises the likelihood of the correspondence, assuming a Gaussian noise model for inliers and a uniform distribution for outliers. Similarly to the algorithm presented in this paper, this is achieved by sampling a set of reference points (the minimum number of points required), estimating the model parameters from them, and rating the parameters estimate using all the points. There are several major differences with this paper though: here, image likelihood and shape likelihood are maximised jointly, apart for the early rejection of step 2, no hard decision is made about correspondences until the full object is detected. Another difference is that here, a flexible shape model is fitted to an image. To the best of our knowledge, there are no reports on using a RANSAC type algorithm with a flexible shape model.

3D shape reconstruction from feature points: Blanz and Vetter [1], similarly to the algorithm presented here, estimate the flexible shape model parameters from the image correspondence of a few set of feature points. The difference with this algorithm is that Blanz and Vetter assume that the projection parameters are known and that the position of the feature point in the image are manually provided. In [1] a MAP estimate of the parameters is sought whereas here, an ML estimate is found. If we were using more reference points and hence estimating the flexible shape model at stage 4 of this algorithm, it would be possible to find the MAP estimate of the shape model parameters, similarly to [1]. However, then, the computational load of the algorithm would be prohibitive, therefore we chose to use as few reference points as possible, similarly to a RANSAC algorithm.

6. Experiments

Training Which points should be used as feature points? To answer this question, we use the following scheme. First we manually select a set of candidate feature points on the reference head. These candidate points are shown on the left of Figure 2. Then, an appearance model is learnt for each candidate feature point. The training sets for the appearance models are built as follows: As training set, we use 871 im-

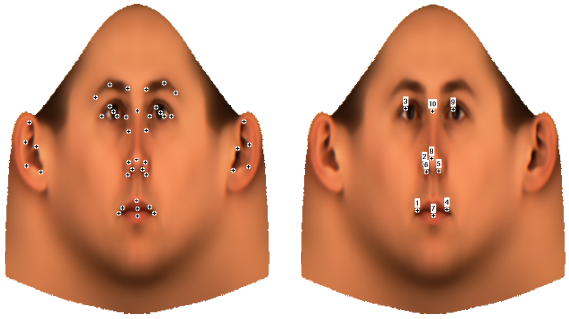


Figure 2. On the right, candidate feature points and one the left, the ones that are selected to be part of the model.

ages from the FERET face image database [15] at frontal and side views. The learning algorithm requires the training facial images to be in correspondence with the candidate feature points. Thus, these images were previously fitted using the Multiple Feature Fitting algorithm [18], thereby recovering dense correspondences between the images used to train the algorithm and 76000 vertices included in the 3DMM. The SIFT operator is then applied to each training images. The points detected by SIFT that are within 5 pixels of the candidate points are included in the training set for that candidate point. For each candidate points, the scale/orientation space is clustered. A GMM is learnt (using an EM algorithm) for each scale/space cluster on the PCA-SIFT features [12]. Then, the appearance models are evaluated on a set of validation images. The feature points are the N_p candidate points with the highest detection accuracy (shown on the right of Figure 2).

Face detection In this section, we test the feature point detection algorithm on a face detection applications. The experiments reported here were carried out on a subset of the CMU-PIE face image database [20] that includes 68 individuals. We used the frontal and side views (cameras 5 and 27) and 6 illumination conditions (flashes 9, 12, 13, 14, 20 and 21). We chose these light directions because the flashes are located near the cameras. Although the shape model (Section 2) can handle any imaging conditions, the appearance model that we use (GMM) is rather simple and we, therefore, limit the testing of the system to these images. Additionally, We use an extensive set of non-face images: more than one thousand images with an average size of 800k pixels composed of the Caltech background, cars, entrances and houses images and a private test set.

The novelty of this paper is the shape model of Section 2. Hence, it is interesting to understand what is the added value of the shape model to the appearance model. This is shown in Figure 3. The left graph is a plot of the histograms of the negative log likelihood of the appearance model , i.e. the output of a classifier that would be solely based on appearance. This is done for the 10 selected feature points. The set of negative examples is a random subset of one million SIFT detected points in the non-face images (red dash line) and the set of positive examples is formed by the SIFT points located within 5 pixels of the ground truth position of the feature points (blue line). As can be seen from the extent of the overlapping region, no clear decision can be taken solely based on local image appearance. These histograms should be compared with the ones on the right plot of Figure 3. These are the histograms of the output of the face detection system for the true positive detections (blue line) and the non-face images (red dash line). The criterion used to decide that detection has succeeded is that the mean localisation error of the visible feature points is within 6% of the inter-eye distance (IED). Recall that the visibility of a feature point is estimated by the algorithm. The magenta slash dotted line is the histogram of the false negatives (face image for which the mean localisation error of

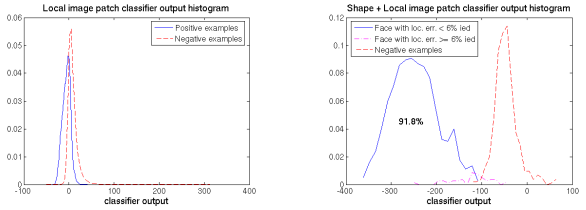


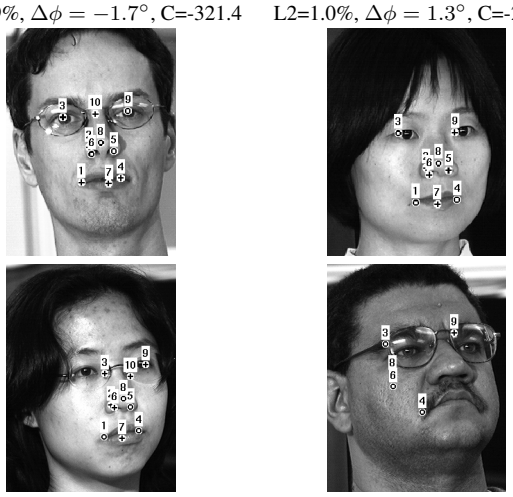
Figure 3. Left: Histogram of the output of the GMM based local patch classifier. Right: Output of the full system. The detection rate is shown for zero false positive. The dash-dotted line shows the output of the false negative: the face images for which the mean error of the feature point is higher than 6% of the IED.

the landmark point is higher than 6% of the IED). Clearly, the overlap between the true positive and the negative examples is almost null. We obtained a 91.8% detection rate with a threshold that achieves *zero false positive*. We are not aware of another face detection system that achieves better result. As an example, one of the state of the art face detector, Vector Boost [11], recently developed, reports that a detection rate of 89.6% is obtained with 72 false positives on the CMU-profile face test set (that use far less non face testing images than the present evaluation). Note that this is indicative only, as both system were not tested on the same data-set: We did not experiment on this CMU-profile set because the resolution of the images is very low: most of the faces are within a square of 40×40 pixels. Our image appearance model was not built to work in this scenario.

Figure 4 shows examples of accurate detection and localisation of feature points (top row) and less accurate or misdetected feature points (bottom row). Accuracy wise, 50% of the feature points are within 4 pixels of the ground truth and 90% within 10 pixels (the distance between the corners of the eyes is on average 130 pixels). The estimated azimuth angle is within 6° for 60% of the face images, and within 20° for 90% of the face images.

Automatic identification The position of the detected feature points can be used to initialise a 3DMM Fitting [18] that yields the 3D shape and texture coefficients of the photographed face. These coefficients can then be used for various tasks such as identification. It is interesting to see whether the accuracy of the feature point localisation is such that the fitting initialised from them would be accurate. To experiment this, we performed an identification on the same set of images as those used for detection. One image per individual is used in the gallery set (the one with the flash light 21, using other illuminations in the gallery set yielded similar results). Rank 1 identification results, averaged over lighting conditions, are provided in Table 1 for different gallery and probe views. They should be compared with those obtained with a manual initialisation [18]. Here, the mean identification rate is 85.6% and is 98.5% using manually provided feature points. In Table 1, the identification results are clearly better for a frontal view than for a side view. There is no similar trend for manually positioned feature points, indicating that this is an artifact from the au-

L2=0.9%, $\Delta\phi = -1.7°$, C=-321.4 L2=1.0%, $\Delta\phi = 1.3°$, C=-243.7



L2=6.3%, $\Delta\phi = 22.4°$, C=-253.8 L2=39%, $\Delta\phi = 51.8°$, C=-41.9

Figure 4. Feature point detection examples. The mean localisation error of the feature points is less than 6% of the inter-eye distance for the top row images and higher than 6% for the bottom row. The mean localisation error, the azimuth angle error and the negative log likelihood ratio are also shown. The bottom left and right images are the ones with, respectively, the lowest and highest classification output among the images with mean localisation error higher than 6%.

tomatic feature point detector. This is probably because about 75% of the training images are frontal.

7. Conclusion

We have presented a novel weak perspective invariant probabilistic 3D shape model and have used it with a feature point detector algorithm. Using the shape model in combination with the detection framework of Burl *et al.* [4] provides robustness to occlusion of feature points and with the conditional independence used in Felzenszwalb *et al.* [7] provides efficient detection.

For a face detection application, it was demonstrated that no false alarm are obtained for a high detection rate despite the utilisation of simple appearance models (GMMs) and the large negative testing set. The detected feature points can be used to initialise a 3DMM fitting algorithm. It is shown that good identification results are obtained from the coefficients estimated by automatic fitting.

In the future, we want to increase the localisation accuracy, the range of poses and the illumination conditions that the appearance model can handle. We plan to replace the GMMs used for the appearance models by an efficient model based on SVM [16]. Ultimately, if the appearance models are discriminative and fast enough, an interest point operator, such as SIFT, might not be necessary.

	frontal probe	side probe
frontal gallery	92.6%	85.8%
side gallery	83.8%	80.0%

Table 1. Identification percentage averaged over illumination conditions. The mean identity percentage averaged also over pose is 85.6%.

References

[1] V. Blanz, A. Mehl, T. Vetter, and H.-P. Seidel. A statistical method for robust 3D surface reconstruction from sparse data. In *Symp. on 3D Data Proc., Vis. and Trans.*, 2004.

[2] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D-faces. In *SIGGRAPH 99*, 1999.

[3] V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. *PAMI*, 2003.

[4] M. C. Burl, M. Weber, and P. Perona. A probabilistic approach to object recognition using local photometry and global geometry. In *ECCV*, pages 628–641, 1998.

[5] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models - their training and application. *CVIU*, 1995.

[6] D. Crandall, P. Felzenszwalb, and D. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *CVPR*, 2005.

[7] P. Felzenszwalb and D. Huttenlocher. Pictorial structure for object recognition. *IJCV*, 61(1), 2005.

[8] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *CVPR*, 2003.

[9] R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *CVPR*, pages 380–387, 2005.

[10] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis. *Comm. of the ACM*, 24(6), 1981.

[11] C. Huang, H. Ai, Y. Li, and S. Lao. Vector boosting for rotation invariant multiview face detection. In *ICCV*, 2005.

[12] Y. Ke and R. Sukthankar. PCA-SIFT: A more distinctive representation for local image descriptors. In *CVPR*, 2004.

[13] K. R. Koch. *Parameter Estimation and Hypothesis Testing in Linear Models*. Springer, 1988.

[14] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2), 2004.

[15] P. J. Phillips, P. Rauss, and S. Der. Feret (face recognition technology) recognition algorithm development and test report. Technical report, U.S. ARL, 1996.

[16] M. Rätsch, S. Romdhani, and T. Vetter. Over-complete wavelet approximation of a support vector machine for efficient classification. In *Proc. DAGM'05*, 2005.

[17] S. Romdhani, N. Canterakis, and T. Vetter. Selective vs. global recovery of rigid and non-rigid motion. Technical report, CS Dept, University of Basel, 2003.

[18] S. Romdhani and T. Vetter. Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *CVPR*, 2005.

[19] H. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *PAMI*, 20:23–38, 1998.

[20] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination and expression (pie) database of human faces. Technical report, CMU, 2000.

[21] P. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *CVIU*, 78, 2000.

[22] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *CVPR*, 2001.

[23] M. Weber, M. Welling, and P. Perona. Unsupervised learning of models for recognition. In *ECCV*, pages 18–32, 2000.