Int J Comput Vis
DOI 10.1007/s11263-016-0967-5

Markov Chain Monte Carlo for Automated Face Image Analysis

Sandro Schönborn[1] · **Bernhard Egger**[1] · **Andreas Morel-Forster**[1] · **Thomas Vetter**[1]

Received: 7 July 2015 / Accepted: 17 October 2016

Abstract We present a novel fully probabilistic method to interpret a single face image with the 3D Morphable Model. The new method is based on Bayesian inference and makes use of unreliable image-based information. Rather than searching a single optimal solution, we infer the posterior distribution of the model parameters given the target image. The method is a stochastic sampling algorithm with a propose-and-verify architecture based on the Metropolis–Hastings algorithm. The stochastic method can robustly integrate unreliable information and therefore does not rely on feed-forward initialization. The integrative concept is based on two ideas, a separation of proposal moves and their verification with the model (Data-Driven Markov Chain Monte Carlo), and filtering with the Metropolis acceptance rule. It does not need gradients and is less prone to local optima than standard fitters. We also introduce a new collective likelihood which models the average difference between the model and the target image rather than individual pixel differences. The average value shows a natural tendency towards a normal distribution, even when the individual pixel-wise difference is not Gaussian. We employ the new fitting method to calculate posterior models of 3D face reconstructions from single real-world images. A direct application of the algorithm with the 3D Morphable Model leads us to a fully automatic face recognition system with competitive performance on the Multi-PIE database without any database adaptation.

Communicated by T.E. Boult.

✉ Sandro Schönborn
sandro.schoenborn@unibas.ch

[1] Department of Mathematics and Computer Science, University of Basel, Spiegelgasse 1, 4051 Basel, Switzerland

Keywords Face image analysis · Markov chain Monte Carlo · Model fitting · Morphable Model · Generative models · Top-down and bottom-up integration

1 Introduction

Understanding images of human faces is among the most important problems in computer vision. Deformable Parametric Appearance Models (PAM) (Blanz and Vetter 1999; Cootes et al. 2001; Matthews and Baker 2004) are a widespread category of methods to solve this task. Most PAMs are used in a generative setup with the intention to reconstruct the input image with a parametrically-controlled synthetic image. Such an Analysis-by-Synthesis approach leaves one with the problem of how to find suitable parameters given an input image (fitting). The fitting problem is difficult to solve. The minimization of the difference between the synthetic image and the target is highly non-convex and usually very high-dimensional ($d > 100$). Most methods use standard optimization algorithms, which require a very good initialization and are prone to local optima. They cannot deal with unreliable initialization and run into trouble if the calculated gradients are not accurate enough.

We propose a novel probabilistic strategy for face model fitting which is based on a Data-Driven Markov Chain Monte Carlo (DDMCMC) (Tu et al. 2005) algorithm. It is stochastic by design and produces a probabilistic result instead of a point estimate. This makes the method suitable for uncertain information as well as more robust towards local optima. Its result is a posterior distribution, including information about the certainty of the model fit.

The method is based on the separation of the optimization iteration into a *proposal* and a *verification* stage. The split removes the need for each update step to strictly improve

Published online: 02 November 2016

the likelihood value and allows the method to also incorporate updates which are misleading. The method is formally based on the Metropolis–Hastings (MH) algorithm, which makes it a sampling-based fitting algorithm. Due to its stochastic nature, the method becomes much less prone to local optima and achieves high-quality reconstructions. Contrary to traditional optimization, we integrate initialization information directly into the Bayesian inference process of fitting. Stochastic filtering, a cascaded application of the Metropolis acceptance rule, generates samples from a sequence of Bayesian conditional distributions.

We present and evaluate the probabilistic fitter with the 3D Morphable Model (3DMM) for face reconstruction from a single image. For this, we reformulate the 3DMM probabilistically and present fitting as finding the posterior distribution of the model's parameters conditioned on the target image. We demonstrate and study the integrative capabilities of Bayesian stochastic filtering by including face and feature point detection directly into inference. The proposed integration enables the method to reconstruct faces fully automatically while avoiding a premature, irreversible decision which is a problem in most feed-forward architectures.

In order to study the remaining uncertainty in a fit, we also present a new collective likelihood to evaluate the degree of fit between the model and the image. It is based on the average squared distance between two images rather than the product of independent Gaussian distributions at each pixel. While the large product leads to extremely sharp posterior distributions due to the many thousand observations, the collective likelihood lets us sample different solutions which show a specified distance to the target image. With the collective likelihood, we can study the remaining uncertainty of a fit. The collective likelihood model is based on the idea of the Central Limit Theorem to approximate a large average value with a Normal distribution even if the individual constituents are not normally distributed. It is thus also more insensitive with respect to the actual noise distribution.

The MCMC fitter does not need gradient information, which makes it a natural match for models which do not provide accurate gradient information, e.g. due to self-occlusion or stochastic elements. But also in general, the adaptation of a parametric face model to an image typically leads to a rough cost function. Optimizing such functions with gradients is difficult without further tricks such as smoothing or multi-resolution approaches.

The results of the probabilistic fitter are useful for a wide range of applications, from shape measurements to face recognition. We evaluate the proposed fitting method with respect to its reconstruction performance and present unconstrained face recognition as a straight-forward application. For face recognition, we only use the general purpose model without any database adaptation. We evaluate on multiple standard datasets, such as the renderings published with the *Basel Face Model*, the *Multi-PIE* database for recognition, the *BU-3DFE* faces set for 3D reconstruction and to some extent *Labelled Faces in the Wild* for an impression of the method's practicality. Besides the applications, we also study the remaining variability of fits as an exclusive result of a probabilistic fitter.

Contribution Our primary contribution is adapting the MH algorithm to support a probabilistic propose-and-verify fitting approach for face fitting. The method produces samples from the posterior distribution and thus a fully probabilistic fitting result. Our method is especially well-suited to integrating unreliable information from different sources and heuristics directly into the Bayesian inference process. From a practical point of view, the benefits of the stochastic method include adaptation without gradients, lower liability towards local optima, information about the certainty of a fit and no need for a feed-forward initialization. We present a novel and more robust approach to fully automatic fitting of a 3DMM to a single image. We also present a new likelihood where we model the distribution of the average squared difference between two images. This stands in contrast to the usual large product of independent individual likelihoods for each pixel value.

Overview The proposed method is a generative Analysis-by-Synthesis model fitter. We try to explain an image $\tilde{I}$ by the parametric model with parameter θ such that the generated model image $I(\theta)$ is close to the target image. We draw samples from the posterior distribution of the model parameters given the target image $P(\theta \mid \tilde{I})$ using the MH algorithm. The algorithm formalizes a propose-and-verify concept where each iteration is split into a proposal and a verification stage. A proposal is simply a parameter update $\theta \rightarrow \theta'$ to the current explanation state θ. The update is unconstrained and can even be random. The quality of the new state is then checked in the verification stage, where we evaluate the likelihood of the proposed parameter $\ell(\theta' \mid \tilde{I})$ and decide whether to keep the updated state or reject it. Instead of using just a single likelihood evaluation, we propose to cascade the verification stage to filtering steps with respect to different likelihoods. We include different kinds of information, e.g. feature point locations, directly into the Bayesian inference process through filtering with the respective likelihood. This makes the algorithm independent of a feed-forward initialization. In Fig. 1, we present an overview of our exemplary implementation for adaptation of the 3DMM to a single image where we integrate face and landmarks detection through cascaded filtering.

In the remainder of the article, we first discuss relevant background information in detail, including deformable parametric models and their standard fitting algorithms. The

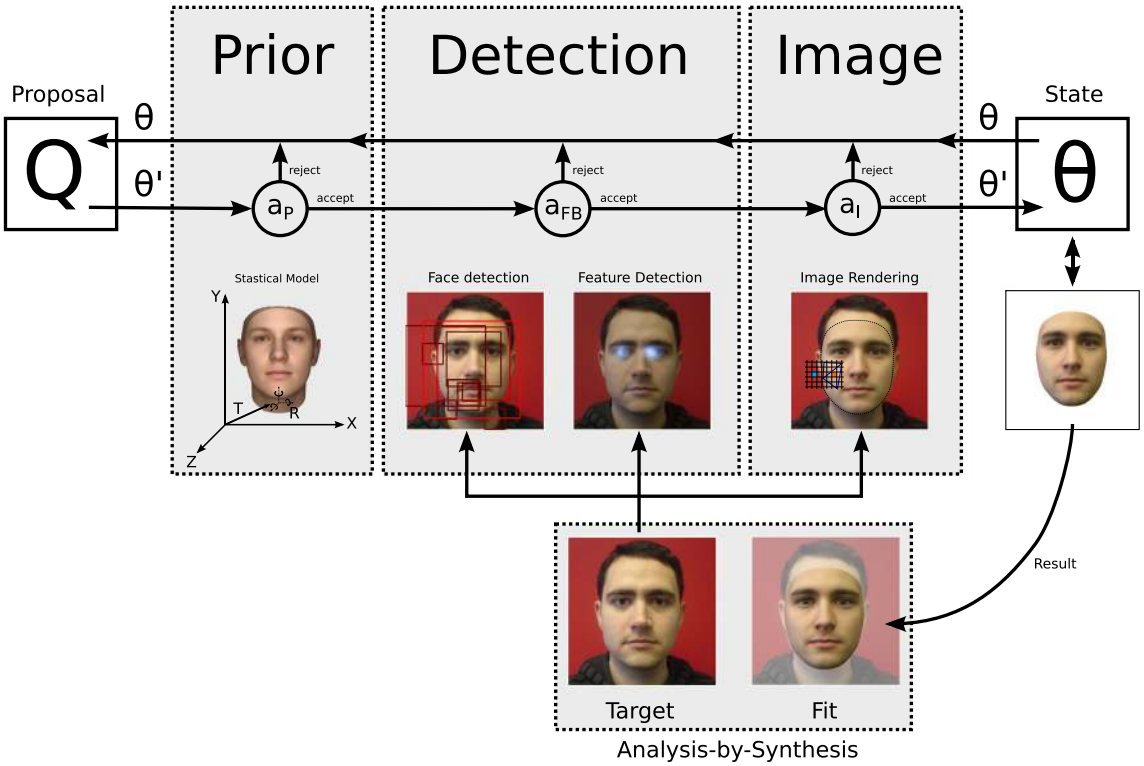

Fig. 1 Overview. The core of our sampling framework is a propose-and-verify architecture. Updates are drawn by proposal generator Q and evaluated in multiple filtering steps. The face model prior, detection results and the target images are used to verify the quality of the proposal. A proposal is accepted or rejected at each filtering stage (a_X) using a stochastic MH acceptance step. The method yields samples of the posterior distribution over our model parameters θ

presentation of the method and our exemplary implementation for adaptation of the 3DMM to a single image consists of four parts: the probabilistic fitting setup in Sect. 3, the proposals in Sect. 4, the applied likelihood models in Sect. 5 and the integrative filtering concept in Sect. 6. In Sect. 7, we evaluate the resulting fitting algorithm in different tasks, such as diagnostic runs of the sampler as well as applications for 3D face reconstruction and face recognition. A discussion with comparisons to existing fitters follows in Sect. 8.

2 Background

2.1 Parametric Appearance Models

Parametric Appearance Models (PAM), such as the Active Appearance Model (AAM) (Cootes et al. 2001) and the 3D Morphable Model (3DMM) (Blanz and Vetter 1999), are a commonly used tool for generative face image analysis and manipulation.

Compared to simpler models which only describe the pixel-based image appearance, such as e.g. eigenfaces (Kirby and Sirovich 1990; Turk and Pentland 1991), a parametric face model also captures the shape of the face through correspondence information. The correspondence is defined through a *reference* face, which is *deformed* to match the geometry of the model instance.

Correspondence information must be obtained for all training samples. It is usually based on user-identifiable reference points, called *landmarks*. The shape part of the model becomes a point distribution model which describes the spatial distribution of the selected points in the training samples. For the 3DMM this process is only the first step, the final model is acquired by performing a dense registration on the whole face, based on the landmark values as weak boundary condition.

A PAM can render artificial images $I(\theta)$, controlled by the values of their parameters θ through

$$I(\theta) = \mathfrak{R}\left(M\left(\theta_S, \theta_C\right); \theta_P, \theta_L\right). \tag{1}$$

The parameter set $\theta = (\theta_S, \theta_C, \theta_P, \theta_L)$ contains all relevant information for the image formation process. It includes a description of the face instance M with shape and appearance (θ_S, θ_C) as well as the image transform parameters θ_P (pose), which control the rendering process $\mathfrak{R}$, mapping the model into the image and finally the illumination setup in θ_L.

In order to generate meaningful images, the distribution of the parameters is matched to some estimated statistics. By far the most common type of representation is the Principal Components Analysis (PCA), e.g. in Cootes et al. (2001) and Blanz and Vetter (1999). To get a real prior distribution on the complete data space, a probabilistic extension through Probabilistic PCA (PPCA) is necessary (Tipping and Bishop 1999; Albrecht et al. 2013), see Sect. 3.1.

2.2 Model Adaptation

A PAM for face image analysis is used in a generative setup. The synthetically generated image is required to match the input image as closely as possible. The model adaptation is then stated as a regularized optimization problem. The parameters $\theta^\star$, which explain the image best, are obtained by minimizing a distance measure C between the model-generated image I and the target image $\tilde{I}$ together with a regularization term R

$$\theta^\star = \arg\min_\theta C\left(I\left(\theta\right), \tilde{I}\right) + \mathcal{R}\left(\theta\right). \tag{2}$$

The choice of cost function C can be motivated by probabilistic considerations. It correspond to $-\log \ell(\theta; I)$, where $\ell(\theta; I)$ is the likelihood of the parameters given the input image. Minimization of cost then corresponds to a Maximum-a-posteriori (MAP) estimator. In practice, cost function and regularization are usually chosen to be sums of squared differences, motivated by Gaussian distributions.

The methods of choice to solve (2) are quite different. They range from (stochastic) gradient descent to highly adapted and efficient compositional methods, even machine learning approaches can be used. All of these iterative solutions calculate a parameter update, based on the current value θ_n and mainly differ in the method to calculate the update f in $\theta_{n+1} = \theta_n + f(\theta_n, \tilde{I})$.

The update $f(\theta_n, \tilde{I})$ is based on the local gradient (stochastic gradient descent, Blanz and Vetter 1999), local linearization of function composition (ICIA or warp-based methods, Matthews and Baker 2004; Romdhani and Vetter 2003) or local quadratic approximation (Romdhani and Vetter 2005). All of these algorithms are built to optimize the squared error cost $C(I, \tilde{I}) = \|I(\theta) - \tilde{I}\|^2$. Detailed and expensive gradient calculations are necessary for reliable operation. Natural images normally exceed the model space of a PAM and show many fine details. Calculated gradients therefore become less representative and increasingly affected by noise. Most methods are not built to deal with unreliable gradients. Due to the local validity of gradients, these methods need rather precise initialization and only reach locally optimal solutions. The initialization is traditionally provided by the user.

The stochastic gradient descent method of the original 3DMM fitter (Blanz and Vetter 1999) is somewhat more forgiving as it is non-deterministic and might recover from a wrong update direction. It is also capable of avoiding some local optima. But the randomness of the method, stemming from a partial evaluation of the full gradient, is arbitrary and lacks a systematic interpretation and analysis.

An alternative approach is taken by Aldrian and Smith (2013). They propose to solve the problem in a strict feed-forward setup. The method relies on many user-provided feature point locations which are used to infer the shape of the face. Once the algorithm has decided on the shape, the illumination and texture are reconstructed using a fixed geometry. The algorithm leads to accurate and fast results if everything is properly set up and all the required points are available. In a setup with uncertain input information and unreliable detection it is dangerous to rely on proper initialization and fix the shape early. A later correction using image appearance information, e.g. shading, is not possible.

The Supervised Descent Method (SDM) (Xiong and De La Torre 2013) uses machine learning methods to predict the update step from the difference between the current state and the target. Through the learning step, the approach is more robust to naturally occurring disturbances such as glasses, beards or face details not present in the model. Contrary to the standard methods above, the SDM minimizes an image feature difference (e.g. SIFT, Lowe 2004) rather than squared pixel differences. Measuring image differences with these more abstract image features should already make the method more robust but makes gradients impossible to calculate. The recently developed method is based on cascaded regression and already quite successful. It has been introduced to adapt 2D models of faces (Xiong and De La Torre 2013) without self-occlusion and complex rendering functions. There is also a recent variant to adapt 3DMMs to frontal views (Zhu et al. 2015). It remains open to this day whether the method is actually suited to explain images with the full flexibility of the 3DMM, such as face pose up to profile side views with illumination and perspective camera setup.

None of the current methods is suitable for probabilistic inference besides a local MAP analysis. All of the methods rely on good update steps and a good initialization. Gradients and Hessians are sometimes very expensive to obtain and easily affected by noise or uncertain input data. Even though SDM methods relieve this problem by learning from actual data, including noise, they still fundamentally rely on good update steps.

We propose to split the problem into two parts, *proposal* and *verification*, where possible updates are only accepted as next steps if they pass the model validation stage. The validation should not be perfectly strict and allow 'backwards' steps towards worse solutions. Both is naturally offered by

the well-known MH algorithm, a representative of Markov Chain Monte Carlo methods.

2.3 Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) methods are a common tool to perform approximate inference with intractable probabilistic models. The posterior distribution is approximated using a set of random samples. All algorithms of the MCMC-type draw samples from Markov Chains where the target distribution is the equilibrium distribution. A good technical overview is provided in Robert and Casella (2004) and Chib and Greenberg (1995) while more practical aspects are discussed in the classic text (Gilks et al. 1996).

We make use of the MH algorithm (Metropolis et al. 1953; Hastings 1970) which builds its Markov Chain by accepting or rejecting samples drawn from a proposal distribution. It is a very general algorithm which is applied to solve a variety of problems. The algorithm is generally used to perform Bayesian inference but also has explicit applications to solve inverse problems, e.g. in geophysics (Sambridge and Mosegaard 2002).

The MH algorithm transforms samples θ', drawn from a proposal distribution $Q(\theta'|\theta)$, into samples stemming from the target distribution $P(\theta)$. The algorithm accepts a proposal as a new sample with probability

$$a = \min\left\{\frac{P(\theta')}{P(\theta)}\frac{Q(\theta|\theta')}{Q(\theta'|\theta)}, 1\right\}. \tag{3}$$

On rejection, the algorithm keeps the current sample θ. Normalization of P is not required, as only ratios of probabilities are considered by the algorithm.

Computer vision models usually consist of many parameters of different scale and interdependence with variable meaning to the image formation process. It is difficult to design a general MCMC sampler in this field without adaptating it to the concrete problem. The more recent development of Data-Driven Markov Chain Monte Carlo (DDMCMC) extends the concept to integrate data-driven proposals (Tu et al. 2005). Such proposals include probably useful knowledge extracted directly from the target image. Fast machine learning methods can be used to construct a more efficient sampler. The methods fit the problems of computer vision much better than pure MCMC methods.

DDMCMC methods are useful to solve large inverse problems with many parameters of varying meaning where some heuristics exist but are not reliable enough to be used on their own. Heuristics are most useful in richly structured problems. A common theme is to use proposals which treat blocks of the model as independent, e.g. objects in scenes. But each of those proposals is always checked with the complete model to ensure consistency among the parts. DDMCMC are applied successfully to segment images (Tu et al. 2005), infer a complex 3D scene from monocular input (Wojek et al. 2010), adapt a human body model to an image (Rauschert and Collins 2012) or to localize faces in images (Liu et al. 2002).

Recently, DDMCMC has been proposed as a general solution to inverse graphics problems, termed the *informed sampler* (Jampani et al. 2015). The authors present an intriguing idea of forming general data-driven proposals using kernel density estimates, but mainly demonstrate the usefulness using small artificial rendering problems. A very similar approach is presented in Kulkarni et al. (2015), where the authors focus on casting the concept into a programming language intended for broad application.

3 Probabilistic Fitting

Our proposed fitting strategy is based on a Bayesian interpretation of image reconstruction. It builds upon a probabilistic face model as a prior and the MCMC sampling strategy for inference. The probabilistic approach does not result in a single point estimate but in samples from the posterior distribution, conditioned on the target image. It allows us to deal with uncertainty and unreliable information of various origins. We build the basic posterior distribution of the image reconstruction problem from a prior $P(\theta)$ and an image likelihood $\ell(\theta; \tilde{I})$

$$P(\theta|\tilde{I}) \propto \ell(\theta; \tilde{I})P(\theta). \tag{4}$$

The reconstruction problem turns into probabilistic inference of $P(\theta|\tilde{I})$. The posterior distribution is intractable to normalize and difficult to optimize. To build our MCMC inference method with the MH algorithm, we only need the unnormalized, point-wise evaluation.

In the following, we introduce and describe our prior model and the basic inference method. Further elements of the algorithm, such as proposals and likelihood models, are discussed in detail later.

3.1 Bayesian Face Model

We work with the publicly available Basel Face Model (BFM) (Paysan et al. 2009) which is based on 200 densely registered 3D face scans. The BFM consists of a statistical model of shape and color.

Face model The model describes a face as a linear combination of example faces in dense correspondence, using an efficient PCA-based representation. We restrict the original face mesh to roughly 30,000 vertices, removing the ears and the throat (Fig. 2). To use the model in a probabilistic

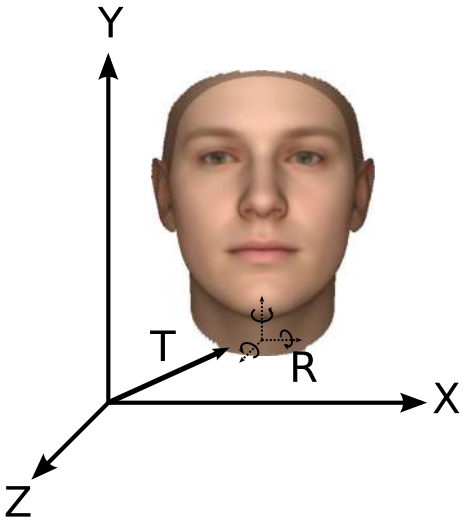

Fig. 2 3D Morphable Model scene setup with the mean face. *Grayed parts* are not adapted to the image

context, we extend it to a Probabilistic PCA (PPCA) model. Such a model is also defined outside the linear span of the training samples and can thus be directly used as a prior distribution of face shape and appearance. This is achieved by adding a spherical Gaussian noise term in the sample space (Albrecht et al. 2013; Tipping and Bishop 1999). Our model then becomes

$$P(\mathbf{x}|\boldsymbol{\theta}) = \mathcal{N}\left(\mathbf{x}|\boldsymbol{\mu} + \mathbf{U}\mathbf{D}\boldsymbol{\theta}, \sigma^2\mathbf{I}\right) \tag{5}$$

$$P(\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{\theta}|\mathbf{0}, \mathbf{I}\right), \tag{6}$$

where we have the mean face in $\boldsymbol{\mu}$, the principal components in matrix $\mathbf{U}$ and the variances along each principal direction in diagonal matrix $\mathbf{D}$. The 3DMM consists of two independent PPCA models, one for shape and one for surface color, which we indicate by the subscripts $()_S$ and $()_C$ respectively.

Scene model To produce an image I, the face is set up in a scene using standard computer graphics rendering. We apply a 3D rotation R and a translation T to align the face relative to the pinhole camera $\mathcal{P}$ (see Fig. 2). A point in 3D is rendered onto the image plane through

$$\mathbf{x}_{2D} = \mathcal{P} \circ T \circ R \circ (\mathbf{x}_{3D}). \tag{7}$$

Illumination model Compared to the original 3DMM setup, we use a different illumination model. A single directional light source is often inappropriate outside lab situations. The face within the scene is illuminated using an efficient representation of both the environment map and the reflectance function through real spherical harmonics basis functions Y_{lm} (Basri and Jacobs 2003; Zivanov et al. 2013).

The complete set of model parameters θ consists of the face representation for shape and color θ_S, θ_C, the scene (pose) description θ_P and the illumination expansion coefficients θ_L. A full value of θ is sufficient to describe a face image of the model.

Further details about the face model and the rendering setup, including estimation of parameters, can be found in Appendix 1.

3.2 Sampling from the Posterior

We propose to move from straight-forward optimization towards a sample-based inference algorithm. The probabilistic result represents the posterior distribution rather than only a maximum. Inference is based on the MH algorithm and therefore produces random samples from the posterior distribution $P(\theta|\tilde{I})$. A sample is generated by first drawing a proposal θ' from the proposal distribution $Q(\theta'|\theta)$. The proposal is only accepted to replace the last sample with probability given by

$$a = \min\left\{\frac{P(\theta'|\tilde{I})}{P(\theta|\tilde{I})}\frac{Q(\theta|\theta')}{Q(\theta'|\theta)}, 1\right\}. \tag{8}$$

Posterior values only appear in ratios. It is therefore sufficient to provide unnormalized evaluation of $P(\theta|\tilde{I})$. The algorithm formalizes a propose-and-verify procedure which is our conceptual tool to deal with unreliable information. Q encodes *proposals* θ' which are possible parameter updates. The algorithm verifies proposals with the posterior distribution P to identify good solutions. To integrate multiple sources of information, we propose a filtering approach which consists of multiple cascaded acceptance stages (see Sect. 6.1 and Fig. 1). The final sampling algorithm consists of many different proposal generators and multiple filtering accept/reject stages.

4 Proposals

The MH inference framework needs proposal distributions to suggest updates for the current parameter values. In this section, we present the basic proposal as a mixture of random walk updates and introduce the illumination estimation.

4.1 Basic Proposal

The algorithm make use of a single proposal distribution Q only. To combine many different proposal distributions Q_i, we build a large mixture distribution

$$Q(\theta'|\theta) = \sum_i c_i Q_i(\theta'|\theta), \quad \sum_i c_i = 1. \tag{9}$$

The mixture coefficients c_i express the probability of drawing a proposal from Q_i. The individual proposals might be unreliable, even completely random. The accept/reject criterion "filters" them to match the posterior $P(\theta|\tilde{I})$. As a basis, we always mix with stochastic random walk proposals. Other parts of the mixture are more informed through the filtering process and serve as data-driven proposals $Q_i(\theta'|\theta, \tilde{I})$. Filtering is described below in Sect. 6.

Random walk proposals The simplest form of a parameter update are random perturbations. Unbiased, they lead to a random walk in parameter space. The random walk proposals are the main source of randomness in the algorithm. To use random walks efficiently, we need to take the different nature of our parameters into account.

The random walk proposal is a mixture of proposals which alter only one of the parameter blocks *camera/pose*, *illumination*, *shape* or *color*. The basic proposal distribution type for a variable in a block b is a normal distribution centered at its current value $Q(\theta_b'|\theta_b) = \mathcal{N}(\theta_b'|\theta_b, \sigma^2)$. We use a mixture of scales (different σ^2) to match the exploration to both rough alignment and detailed adaptation. Updates are multivariate where appropriate, e.g. for shape and color.

Details about the individual blocks and mixtures, including distribution parameters, are presented in Appendix 2.

4.2 Informed Proposals

Proposals are random samples drawn from a probability distribution. Adding deterministic moves can speed-up convergence but also introduce a bias. They still fit the propose-and-verify framework very well, but they need to be mixed with random walks to add a bit of uncertainty. Another solution to circumvent the problem is to restrict usage of deterministic proposals to an initial burn-in phase.

Illumination estimation Our most prominent deterministic proposal is a direct estimation of illumination. Of all model parts, illumination has the strongest effect on pixel intensities. A wrong illumination dominates every other source of image difference.

The light model is linear for a fixed geometry and face color. We can solve for the unknown illumination coefficients while keeping the color and geometry of the face constant (see Appendix 1 for details). We restrict the estimation to only a small random subset of all vertices because illumination does not change on a small scale. The solution is a noisy approximation due to sub sampling and the non-perfect correspondence. The estimation therefore still contains a stochastic element. The proposal is most effective during the beginning, when it leads to promising regions of the parameter space quickly. In later phases of the run, illumination exploration is dominated by random walk proposals.

5 Verification

To perform inference of the posterior distribution $P(\theta|\tilde{I})$, given a target image $\tilde{I}$, we need a likelihood function $\ell(\theta; \tilde{I})$. Likelihoods are necessary in the verification step of the algorithm where they measure the quality of an explanation. In this section, we discuss different choices of likelihoods for images, a widespread product approach and a new but more appropriate collective view. We also introduce the likelihood models necessary to integrate landmarks as well as face and feature point detection.

5.1 Landmarks Likelihood

Fitting the model to observed landmarks is the most common method to align a face model with an image. The 3DMM renders the locations of facial landmark points in the image through (7). The points are observed under a noise model. We model the likelihood with respect to N_{LM} observed landmark positions $\{\tilde{\mathbf{x}}_i\}_{i=1}^{N_{\text{LM}}}$ with independent Gaussian noise

$$\ell(\theta; \tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \ldots, \tilde{\mathbf{x}}_{N_{\text{LM}}}) = \prod_{i=1}^{N_{\text{LM}}} \mathcal{N}(\tilde{\mathbf{x}}_i \mid \mathbf{x}_i(\theta), \sigma_{\text{LM}}^2). \tag{10}$$

We use 9 easily identifiable facial landmarks (see Fig. 4).

5.2 Face and Feature Point Detection

We explicitly integrate Bottom-Up information from face and facial feature point detectors. They are traditionally only used to initialize the model through a rough alignment with the detected face box and the single most certain landmark detections. The downside of integration by initialization is an early decision which cannot be corrected later. Face detection works relatively well, even for strong pose variations and occlusion. However, spurious false positives are still common and can lead to a wrong initialization of the face model fit in feed-forward architectures.

Given an image, we consider the 10 highest-rated face detection candidates. Each of these "face boxes" B_i gives rise to a likelihood $\ell_{\text{B}}(\theta; \text{B}_i)$ which compares the location and scale of the face with the candidate values. We model a positive face detection result as a face box with position $\mathbf{p}_i$ and size s_i. We compare a model instance with the box using a likelihood which combines a log-normal distribution on the scale s and a Gaussian on the position p

Fig. 3 Detection results: all ten candidate face boxes, *colored with brightness* according to certainty (*left*). The left inner eye corner (*middle*) can be detected with a high quality output while the detection result of the right lip corner (*right*) is much more distributed (detection certainty overlaid with *bright blue color*) (Color figure online)

$$\ell_B(\theta; \mathrm{B}_i) = \mathcal{LN}\left(s(\theta) \mid s_i, \sigma_{\mathrm{bs}}\right) \mathcal{N}\left(p(\theta) \mid \mathbf{p}_i, \sigma_{\mathrm{bp}}\right). \quad (11)$$

Feature point detection results are used together with their confidence values, given in a response map $\mathrm{D}_l(\mathbf{x})$ (Fig. 3). The map captures the detector's certainty of seeing landmark l at location $\mathbf{x}$ in the image. Additionally, we also need a landmarks likelihood $\ell_{\mathrm{LM}}(\theta; \tilde{\mathbf{x}})$ which measures the degree of fit to a given feature point location $\tilde{\mathbf{x}}$.

For each face detection candidate i, each feature point detector l delivers a detection certainty map $D_l^i(\mathbf{x})$. To account for imperfect detectors, we additionally limit the maximal certainty values corresponding to false-positive and false-negative probabilities of 0.001.

We construct the likelihood of a landmark falling on location $\mathbf{x}$ to be the best possible combination of detection and distance from the respective model point using our landmarks observation model (10)

$$\ell_{\mathrm{LM}}(\mathbf{x}; \mathrm{D}) = \max_{\mathbf{t}} \mathcal{N}\left(\mathbf{t} \mid \mathbf{x}, \sigma_{\mathrm{LM}}^2\right) \mathrm{D}(\mathbf{t}). \quad (12)$$

The value is precomputed for each location $\mathbf{x}$ using the efficient method from Felzenszwalb and Huttenlocher (2012). Precomputation of the maximum convolution is possible since landmark detections do not change during model adaptation and the landmark certainty is not varied.

The likelihood of each full face candidate, including the box and all l feature point detection maps is then

$$\ell_i(\theta; \mathrm{B}_i, \mathcal{D}_i) = \ell_B(\theta; \mathrm{B}_i) \prod_l \ell_{\mathrm{LM}}(\mathbf{x}(\theta); \mathrm{D}_l). \quad (13)$$

The likelihood including all individual face candidates is constructed as a maximal value

$$\ell_{\mathrm{FB}}(\theta; \mathcal{B}, \mathcal{D}) = \max_i \ell_i(\theta; \mathrm{B}_i, \mathcal{D}_i). \quad (14)$$

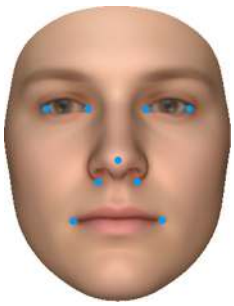

Fig. 4 Facial feature points we detect, drawn on the mean face of the BFM using our face mask. The size of the points corresponds to the standard deviation σ_{LM} of the landmarks likelihood

Choosing a maximum value corresponds to selecting the best possible face candidate i for each parameter value θ. Note that the best candidate i can be different for each θ (Fig. 4).

5.3 Product Likelihood

For full image reconstruction, we also need a model for the likelihood of the target image under a model instance. The standard approach in the Analysis-by-Synthesis setting is an independent, pixel-wise comparison between the rendered image $I(\theta)$ and the target. We pose it as a probability distribution of possible images and evaluate it for the target image $P(\tilde{I}|\theta)$ for a given parameter value θ.

We assume pixel-wise conditional independence and evaluate in the target image, within the region of the rendered face FG

$$P(\tilde{I}|\theta) = \prod_{i \in \mathrm{FG}} P(\tilde{I}_i | I_i(\theta)) = \prod_{i \in \mathrm{FG}} \ell(\theta; \tilde{I}_i). \quad (15)$$

Figure 5 gives a schematic overview of rasterization and the notion of foreground and background.

We assume independent Gaussian noise all over the foreground face region

$$\ell_{\mathrm{FG}}(\theta; \tilde{I}_i) = \frac{1}{N} \exp\left(-\frac{1}{2\sigma^2} \left\| \tilde{I}_i - I_i(\theta) \right\|^2\right). \quad (16)$$

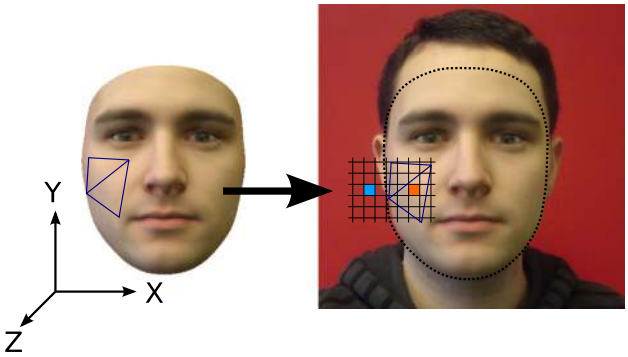

Fig. 5 The 3DMM is projected onto the image plane, where we perform a rasterization of each triangle of the 3D model. Image pixels which lie within the projected face region (*orange*) are considered foreground while those outside (*blue*) are background (Color figure online)

The choice of likelihood roughly corresponds to the usual sum of squared differences. Due to truncation, there are minor differences through normalization if the rendered model color is close to the limits of the intensity range (see Appendix 1).

Background model We evaluate (15) in the image domain. To prevent a shrinking of the foreground region, a background model ℓ_{BG} is necessary. We apply the foreground correction mechanism presented in Schönborn et al. (2015)

$$\ell(\theta; \tilde{I}_i) = \frac{\ell_{\mathrm{FG}}(\theta; \tilde{I}_i)}{\ell_{\mathrm{BG}}(\tilde{I}_i)}. \tag{17}$$

Different background models are discussed in detail in above reference. We namely make use of the constant (constant likelihood ℓ_{BG}) and the histogram background model.

5.4 Collective Likelihood

The product likelihood assumes independent normal distributions at every location of the image. This measure is suited to find a single maximally good fit with the least amount of deviation per pixel. But it depends on the amount of pixels used for image comparison. In practice, evaluation of image difference is often based on averaged measures, such as the mean squared error. The collective likelihood allows us to use average differences in the likelihood and to explore multiple solutions which match the target image at a specified noise level.

The collective likelihood model is based on the fact that large sums of independent values with bounded variance obey a Central Limit Theorem (CLT), e.g. Gonick and Smith (1993). The large sum has a natural tendency to approach a normal distribution, even if the individual constituents are not normal. The assumption of independence among the individual pixels is still required for applying the CLT.

Table 1 Parameters for our collective likelihood model (21) arising from various distributions of the residuals $\mathbf{d} = (d_R, d_G, d_B)$

Model	d^2	$\mathrm{E}[d^2]$	$\mathrm{V}[d^2]$
$d_{\{R,G,B\}} \sim \mathcal{N}(0, \sigma^2)$	$\Gamma(3/2, 2\sigma^2)$	$3\sigma^2$	$6\sigma^4$
$\|\mathbf{d}\| \sim \mathrm{Exp}(\alpha)$	Weibull$(\alpha, 1/2)$	$2\alpha^2$	$24\alpha^4$
$\|\mathbf{d}\|^2 \sim \mathrm{Exp}(\alpha)$	$\mathrm{Exp}(\alpha)$	α	α^2
Empirical	–	0.072^2	0.0002

Note that we model the squared residuals $d^2 = \|\tilde{I}_i - I_i\|^2$ in the collective likelihood. The gamma distribution $\Gamma(k, \theta)$ is parametrized by shape k and scale θ

The average squared distance between the target and the rendered model image is

$$\overline{d^2} = \frac{1}{N}\sum_{i=1}^{N} d_i^2 = \frac{1}{N}\sum_{i=1}^{N} \left\| \tilde{I}_i - I_i \right\|^2, \tag{18}$$

where the sum is over N pixels inside the face. The value of the squared residuals d_i^2 is bounded by the fact that intensity and RGB color channel values lie within [0, 1]. The distribution of the average $\overline{d^2}$ can be approximated by a normal distribution if N is large (in our case $N > 10{,}000$)

$$\sqrt{N}\left(\overline{d^2} - m\right) \to \mathcal{N}(0, v), \tag{19}$$

$$m = \mathrm{E}[d_i^2],\ v = \mathrm{V}[d_i^2]. \tag{20}$$

This motivates our collective likelihood

$$\ell(\theta; \tilde{I}) = \mathcal{N}\left(\overline{d^2}(\theta) \,\middle|\, m, \frac{v}{N}\right). \tag{21}$$

The parameters m and v can be calculated from theoretical assumptions about d_i^2 or simply be estimated empirically. The assumption of independent Gaussian noise leads to a χ^2_N distribution of $\overline{d^2}/(3\sigma^2)$. Other distributions of d^2 lead to different values of the parameters, see Table 1. In our experiments we use empirically estimated values.

The collective likelihood considers model instances together with the actual noise instantiation. A perfect fit has a low likelihood since the chance of observing many independent variables with a zero noise instantiation is very low. The highest likelihood scores are assigned to solutions which match the image at the specified noise level. With the collective likelihood, we can sample many solutions which show a similar average difference to the target image. It allows us to explore the variability of face fits at a given noise level (see Sect. 7.2).

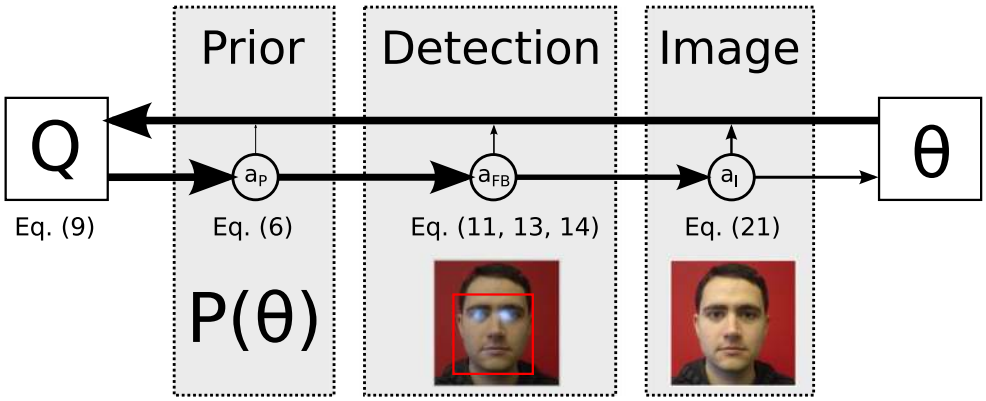

Fig. 6 Information from different sources is integrated into the inference process by filtering. The current state of the sampler θ is updated with a proposal drawn from Q only if it passes all likelihood filter steps. The filters ensure that the proposed sample fits the model's prior distribution, the face and feature point detection maps (FB) and finally the image likelihood. Each filtering stage applies a stochastic MH acceptance step (a_X) with the respective likelihood (the respective equations are indicated). The *thickness of the arrows* corresponds to the amount of samples which are accepted or recjected (see Table 2). For details refer to Sect. 6.1

6 Integration

We propose to incorporate information from different sources, e.g. face and feature point detection, into a single fitting framework by filtering with Metropolis acceptance steps. The cascaded application of the acceptance step of the basic algorithm corresponds to individual steps of Bayesian inference. It provides a fully probabilistic and flexible way to integrate information of various origins, including its uncertainty. Concretely, we demonstrate how to integrate information from face and feature point detectors.

6.1 Integration by Filtering

We integrate additional information, e.g. landmark positions and face detections, by biasing the random walks through filtering. A proper integration into the inference algorithm needs a formulation as a proposal distribution which can generate samples in the parameter space of the model. We do not use direct encoding of feature point positions in the model. Therefore, we have to resort to a generative type of inclusion using our likelihood models.

An integration of (14) into a large product of likelihoods, including all bottom-up parts as well as the final image likelihood $\ell_I(\theta; \tilde{I})$, is not flexible enough. We would have to evaluate the full product for each proposed sample. Instead, we propose to approach the problem with a sequence of Bayesian inference steps where each stage uses the posterior of the previous one as a prior distribution. Such sequential inference is more flexible and suits the propose-and-verify algorithm very well:

$$P(\theta) \xrightarrow{\ell_{\mathrm{FB}}(\theta;\mathcal{B},\mathcal{D})} P(\theta \mid \mathcal{B}, \mathcal{D}) \xrightarrow{\ell_I(\theta;\tilde{I})} P(\theta \mid \mathcal{B}, \mathcal{D}, \tilde{I}). \tag{22}$$

Metropolis filtering We implement each inference stage as a separate Metropolis acceptance filter where each step biases the sample distribution with its likelihood. Implementation as a step-by-step process allows us to drop bad samples early. The filtering approach is very flexible and allows us to integrate almost any knowledge expressed as a likelihood in a simple and canonical fashion.

Starting from the current state θ, we generate a proposal θ', drawn from Q. The proposal is fed through a chain of cascaded stochastic Metropolis acceptance decisions (filters) a_0 to a_n, with

$$a_0(\theta, \theta') = \min\left\{\frac{P(\theta')Q(\theta \mid \theta')}{P(\theta)Q(\theta' \mid \theta)}, 1\right\} \tag{23}$$

$$a_f(\theta, \theta') = \min\left\{\frac{\ell_f(\theta')}{\ell_f(\theta)}, 1\right\}. \tag{24}$$

The first step a_0 generates samples from the prior $P(\theta)$ while each subsequent filter step a_f generates samples from the posterior including likelihoods ℓ_1 up to ℓ_f. A likelihood ℓ_f measures compliance with a datum D_f, which can be anything, e.g. a feature, landmarks, face boxes or image color values. A chain of these decisions can reject a proposal early without evaluating all likelihoods. The proposal is only accepted as new state if it passes all the stages. The posterior distribution fulfills the detailed balanced condition of the resulting transition kernel

$$P(\theta|D_1, D_2, \ldots, D_n) \propto P(\theta)\ell_1(\theta)\cdots\ell_n(\theta). \tag{25}$$

In each stage, we only need likelihood ratios to decide on the fate of a sample, normalization is not required. Addition-

ally, we can extract samples from the respective posterior at each intermediate step.

Integration of detection information For each face box, samples from the unbiased prior distribution are filtered using the face box likelihood thereby biasing them to respect the face box B_i's position $\mathbf{p}_i$ and size s_i. For each landmark, we do the same using the detection map likelihoods. For an overview refer to Fig. 6.

The procedure to draw a single sample then becomes

1. draw a proposed sample from $\theta' \sim Q(\theta' \mid \theta)$
2. apply first acceptance step a_0 with prior $P(\theta)$
   on reject: discard θ', keep θ
3. apply acceptance step $a_{\ell_{\mathrm{FB}}}$
   on reject: discard θ', keep θ
4. apply acceptance step $a_{\ell_{\mathrm{I}}}$
   on reject: discard θ', keep θ
5. update $\theta \leftarrow \theta'$

Note that on any reject, we discard the proposal completely, keep the current sample θ and start over.

6.2 Initialization

To find a suitable starting configuration, we construct Markov chains for each candidate box B_i and draw a few samples, where we use the respective likelihood ℓ_i (13) as a target. After roughly 500 samples, we find parameter regions which correspond to more or less consistent explanations of the feature point detection maps for each detection candidate. The face candidate with the highest consistency and landmarks detection likelihood value is chosen as a starting point for the complete fitting chain.

We use the combined likelihood (14) to evaluate detection consistency during the sampling run. This allows the chain to evaluate with respect to all detection candidates and switch to different explanations if these are more compatible with the image.

7 Evaluation

The evaluation section contains quantitative and qualitative experimentation on multiple databases. The experiments are set up around the problem of 3D face reconstruction and head scene recovery from a single image. The main result is a fully automatic reconstruction of the 3D face from a real-world target image. Depending on the task at hand, additional user-provided or automatically detected landmarks are available besides the target image. We evaluate our method on synthetic, controlled and real-world facial images.

This section starts with specific evaluations of the Markov Chain sampling algorithm in Sect. 7.1. The difference between the collective and the product likelihood is presented in Sect. 7.2. The algorithm results in a posterior distribution which we demonstrate to be a powerful tool for studying the output of the fitter in Sect. 7.4 where we analyze the certainty of our parameters in a fit.

For evaluations concerning the 3D shape recovery (Sect. 7.3), we make use of synthetically generated target images to have a known ground-truth. These cases are explicitly labeled as such and result from an application of our rendering engine, which is also used as part of the generative 3DMM. Note that these images are not model instances. They are based on captured real shape and color of the face (BFM scans) and additionally include a real illumination (BU-3DFE).

The overall application of completely automatic and generic face recognition is presented in Sect. 7.5. The last subsections contain a qualitative evaluation of face reconstruction on real-world imagery.

We compare our proposed sampling method to standard fitting methods applied to the 3DMM throughout this section, namely Aldrian and Smith (2013) and Romdhani and Vetter (2005). Additionally, we present a conceptual comparison in the discussion in Sect. 8.3.

General setup In applications where a single solution is desired, we pick the sample with the highest posterior rating while we use many samples when studying distribution properties. In the latter case, we discard the first part of the run as *burn-in* samples. If not stated otherwise, we draw 10,000 samples from the constructed Markov Chain. In runs where a single optimization-like result is necessary, we use the product likelihood (15, with $\sigma = 0.046$) where we switch to the collective likelihood (21) for real sampling experiments using the empirically estimated parameters from Table 1. The empirical estimation is based on an average reconstruction error of selected very good fitting results. We consider this residual to be an estimate of difference between our model and the world. Expecting solutions closer to the real image is not realistic.

We use two different background models, the histogram model for the recognition experiment and in-the-wild tests and a constant value at two standard deviations of the foreground likelihood model for the other experiments. Both models are discussed in Schönborn et al. (2015).

7.1 Markov Chain Diagnostics

Markov Chain methods can be problematic if one is interested in exact samples from the posterior. Because the chains are built to have the target distribution as equilibrium distri-

Table 2 Acceptance rates

Filter	AR (%)
Prior filter	92
Landmarks filter (user)	75
Detection filter (detection)	71
Collective likelihood (image)	39

Filters show a high acceptance rate, which means only few samples are dismissed in filter stages. The overall acceptance rate with the collective likelihood is within the desired range of 25–50%

bution, we have to wait 'long enough' until we can consider samples as being drawn from the posterior distribution. Since we apply the MCMC method mainly to go beyond a classical fitter and use much of the power of DDMCMC methods to integrate different cues, we are not very rigorous in the diagnostic part. We are satisfied with some basic indicators for a sampling chain rather than an optimizing chain. In the experiments with only a single best result, we do not use diagnostic methods but simply set the number of samples to be drawn to a fixed value of 10,000 samples which yields a satisfying quality in practice.

We performed the following diagnostic experiments on the target image depicted in Fig. 3. To analyze the chain behavior, we drew 100,000 samples from three independent chains.

Acceptance rates The most simple diagnostic is the acceptance rate. If too few samples are accepted, the sampler tries to make moves which are too large, and too many accepted random walk samples indicate that the sampler could walk further in a single step. For general random walk applications, acceptance rates of 25–50% are usually considered acceptable (overview in Chib and Greenberg 1995). We use a filtering strategy to integrate various information into the inference process. A critical point about such filters is a possibly high rejection rate because a sample has to pass all stages.

The acceptance rates of the diagnostic run can be found in Table 2. We observe high acceptance rates for all filters and a total acceptance rate within the desired range.

Samples For further analysis of the sampling chain, we present the sequence of selected sampled parameter values for yaw and a shape parameter in Fig. 7. Additionally, we added the log value of the unnormalized posterior probability (Fig. 8). There is an initial convergence phase of a few thousand samples from very bad posterior values around the starting point towards the region where the model matches the image. In this region, the chain starts sampling and explores according to the posterior distribution's width. If we only use a single best result, we usually stop after the initialization phase of 10,000 samples.

7.2 Collective Likelihood

We constructed the collective likelihood to specify a desired noise model for image reconstruction. We compare this model to the standard product likelihood of independent Gaussians in an image reconstruction task. We draw 50,000 samples from the image reconstruction posterior given the image in Fig. 3 and setup both likelihoods to represent the same noise model: independent Gaussian noise with the same empirically estimated standard deviation ($\sigma = 0.042$). For comparison, we also present results from a second run where we chose a broader product likelihood ($\sigma = 0.058$), see Fig. 9.

The collective likelihood is useful to draw samples from the many different solutions which all lead to an image

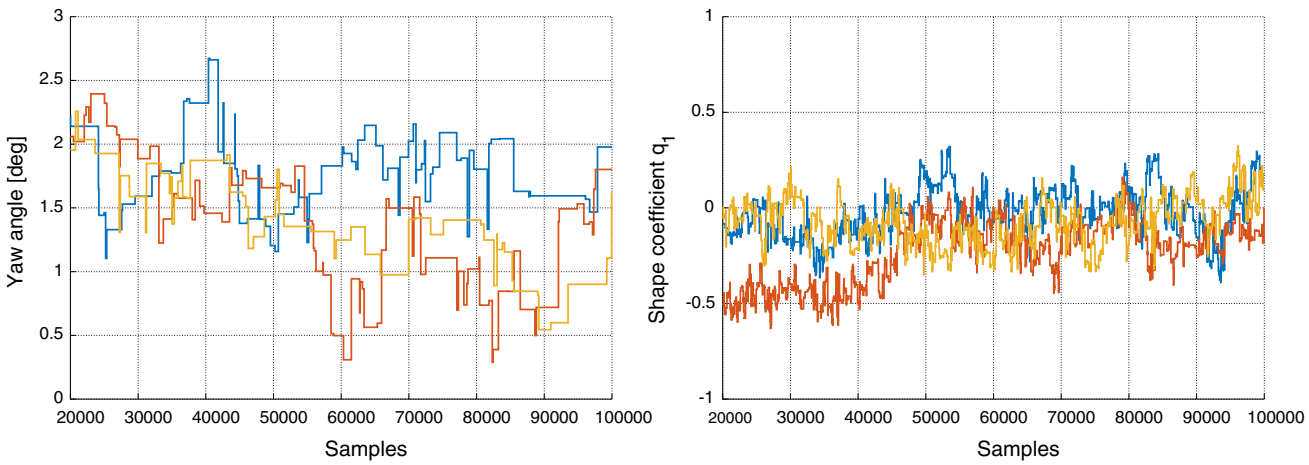

Fig. 7 Sampled parameter values of yaw angle and a shape model parameter for three independent chains with target image from Fig. 3

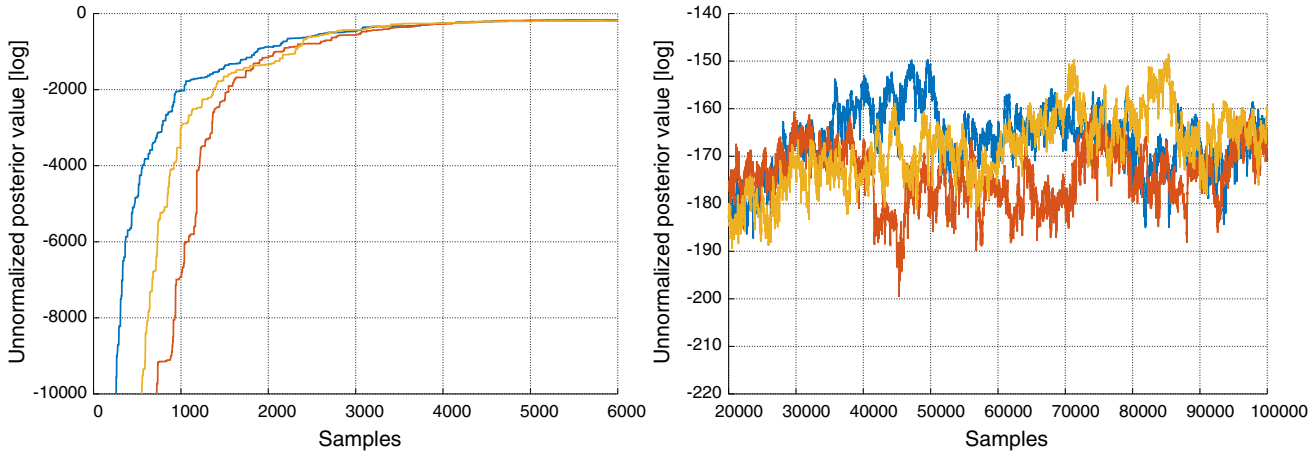

Fig. 8 Unnormalized posterior values of samples drawn during initialization (*left*) and sampling phases (*right*). After a few 1000 samples all three chains converge to the same region of the posterior and explore it appropriately

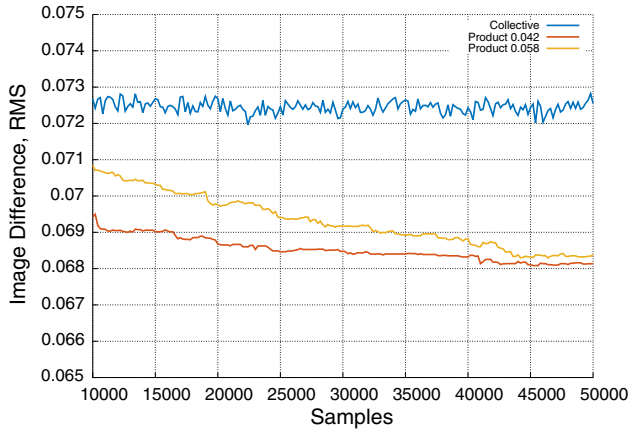

Fig. 9 Average image difference per pixel (RMS) between the model and the target image. We compare samples drawn from posteriors with respect to the product and the collective likelihood with the same noise distribution (Gaussian noise with $\sigma = 0.042$) and a broader product likelihood with 0.058 expected deviation per channel ($\sqrt{\mathrm{E}[d_i^2]} = 0.1$ total per pixel). With the collective likelihood, we can draw samples which all lead to a similar degree of image reconstruction, at the specified noise level ($\sqrt{\mathrm{E}[d_i^2]} = 0.072$). The product likelihoods lead to very sharp posteriors with strong optimization behavior. They converge to the explanation with the least noise necessary to reconstruct the image

explanation of similar quality. These samples represent the posterior variability at the specified noise level. The product likelihood leads to a strongly peaked distribution where sampling shows optimization behavior of ever-increasing fit. The posterior variance almost vanishes and the samples are not suitable to estimate any posterior properties (see Table 3). The final level of image difference does not relate to the noise magnitude. The runs aim for the classical best fit in terms of image difference.

Table 3 Posterior standard deviation of samples from the collective and product likelihoods

Posterior	Yaw (deg)	Shape q_1
Collective	0.34	0.13
Product $\sigma = 0.042$	<0.05	<0.01
Product $\sigma = 0.058$	<0.05	<0.01

The product likelihood does not lead to useful posterior samples. The collective likelihood leads to posterior samples which reflect the remaining variability with a given noise model. The collective likelihood corresponds to the product with $\sigma = 0.042$

7.3 Reconstruction of 3D Face Shape

The main application of the 3DMM and the presented fitting machinery is a complete reconstruction of the dense 3D shape of a face to establish full correspondence between the model and the target image. Using this information, many tasks can then be built on top of this result. Therefore, we try to evaluate the 3D reconstruction quality first. This task is inherently difficult to perform as there is currently no known metric which is close to human perception of face similarity. Therefore, we restrict ourselves to a simple root-mean-square distance (RMSD) with all its shortcomings.

We reconstruct rendered frontal images from the BU-3DFE dataset (Yin et al. 2006). This dataset contains both shape and appearance information. The 3DMM does not contain any expression variability, therefore we restrict this analysis to neutral versions of all 100 individuals in the dataset. We run our fitting method on the plain frontal views of the textured meshes and user-provided landmarks information to obtain the optimal reconstruction performance.

Table 4 3D reconstruction accuracy in mm

Dataset	RMSD (mm)
BFM scans	3.78
BU-3DFE (3DMM)	5.39
BU-3DFE (mean-only)	6.79

The last line is the result of adapting the mean face to the target image, not allowing any changes in the shape and appearance of the face

To obtain an estimation of the reconstruction quality, we render a depth image of both our reconstruction and the original scanned 3D face into the image. Since we do not use a calibrated setup and cannot estimate depth reliably in frontal views (see below), we allow the absolute distance from the camera (t_Z) to vary and optimally align the original and our reconstruction with respect to this value only before calculating the RMSD.

We perform the same reconstruction experiment using ten face scans published for evaluation purposes together with the BFM (Paysan et al. 2009). We use four different scene settings to render the face scans, a frontal view, a side view at 30° yaw and two difficult, realistic illumination situations. The result presented in Table 4 are RMS averages over all four setups. Figure 10 displays renderings of the BFM scans.

BFM renderings To compare our results to other fitting algorithms we also performed the experimental setting proposed by Aldrian and Smith (2013). We reconstruct the 270 synthetic renderings which are provided with the BFM (Paysan et al. 2009). To make the experiment comparable with the literature, we present mean squared errors on the complete face (no mask) where we average per pose and illumination setup but not per vertex and convert to micro meters. Because we cannot reliably determine the distance from the camera (see Sect. 7.4), we rigidly align our shapes with the target before evaluation.

The shape estimation error is compared to the state of the art fitting algorithms in Fig. 11. For comparison reasons, we also use landmarks as provided with the BFM renderings. We reach very similar performance to the multi-feature approach by Romdhani and Vetter (2005). Using the full set of 70 landmarks (Farkas set) we clearly outperform (Aldrian and Smith 2013) which relies on this amount of landmarks as input. However, the authors use a simpler camera model which does not allow for a fully precise reconstruction.

7.4 Posterior Variability

In contrast to other methods, the probabilistic sampling method delivers an estimate of the posterior distribution of the model parameters given a target image. The distribution contains information about the certainty of the fitting result. We extract this information in terms of the posterior variance of the face shape and pose parameters which is present in the samples from the posterior. It expresses the remaining variability after adapting to the target image. This information is very useful as a diagnostic tool concerning individual fits and for deciding on further model improvements.

For these experiments, we fit the model to the complete set of the renderings provided with the BFM (Paysan et al. 2009). We fit the model to two different landmark sets, namely the anchors provided with the renderings (10 points) and the Farkas set (70 points). We then drew 100,000 samples and selected every 25th posterior sample from the second half of the run. The sparsity of this set reduces burn-in effects and correlation between samples. For the evaluation, we measured the posterior standard deviation of face shape and selected pose parameters (Fig. 12; Table 5). For a visual comparison, we rendered the standard deviation of the surface (shape) at each point into a frontal 2D view, see Fig. 12.

The method is able to find the posterior distribution given only a single image. To highlight this, we also present the posterior variability on an individual real-world target image (from Fig. 3). We compare different fitting setups where an increasing amount of information becomes available (see Fig. 13 and Table 6).

In all experiments, the certainty of the model's posterior distribution increases with more or more certain informa-

Fig. 10 A scan available with the BFM rendered in four different scenes, one pose variation and two illumination setups. These views of all ten scans are targets in the 3D shape reconstruction experiment, see Sect. 7.3

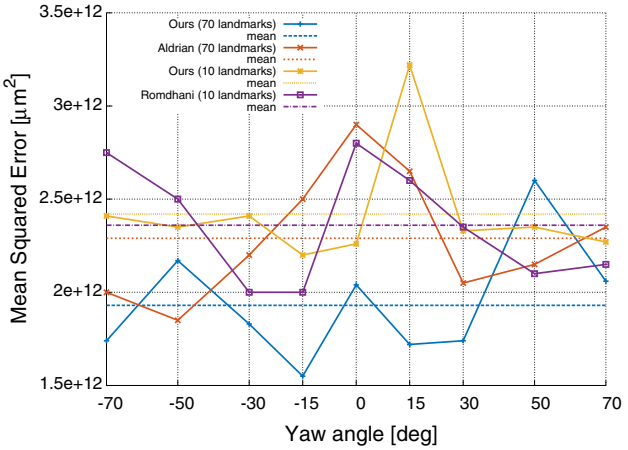

Fig. 11 Shape estimation accuracy on the BFM renderings (Paysan et al. 2009) compared to state of the art fitting methods (Aldrian and Smith 2013) and (Romdhani and Vetter 2005). All numbers are from Fig. 4 in Aldrian and Smith (2013)

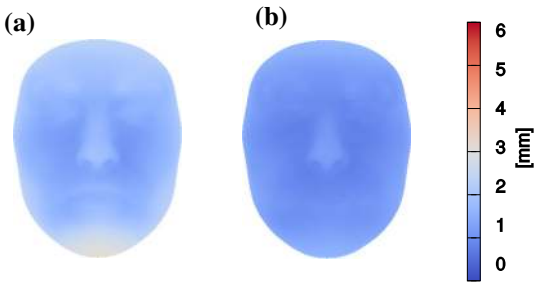

Fig. 12 Posterior variability of the face shape using landmarks as input. Image **a** shows still a rather large variance of the shape while the additional knowledge in **b** constrains the posterior. Average values (RMS) on the BFM renderings. **a** 10 landmarks, **b** 70 landmarks

Table 5 Average (RMS) posterior standard deviations on BFM renderings (Paysan et al. 2009) using 10 landmarks or 70 landmarks (Farkas) as provided with the BFM renderings

Posterior	ϕ (yaw) (°)	t_Z (distance) (mm)
10 Landmarks	1.7	2590
70 Landmarks	0.9	534

tion. The variance of the prior represents the complete model flexibility without any observations. Next, we sample from the posterior distribution conditioned on our detection maps. This distribution already makes a clearer statement about the shape of the face, while the pose is still uncertain. When we add user-provided landmarks with high accuracy ($\sigma = 4$ pixels, 3% inter-eye distance, see Fig. 4), the pose estimation becomes more certain. Conditioning on the image only can restrict the pose with high certainty but not the shape. When conditioning on the image and either detection maps or landmarks, the posterior is most certain, both for pose and shape (Fig. 13; Table 6).

Pose estimation We performed a pose estimation experiment on the BFM renderings (Paysan et al. 2009). 270 renderings over 9 poses and 3 illumination settings are provided in the dataset and used for this experiment. The estimated yaw angle is compared to the available ground truth data. In Fig. 14, we show the yaw estimation error and the standard deviation of the estimated posterior. This experiment shows that a lower standard deviation of the posterior correlates with a higher accuracy of the estimation.

Besides the yaw angle, we also estimate the camera distance. Distance from the camera is very difficult to estimate from a single view. A change of distance mainly leads to a scaling of the image and only to some extent to perspective distortion. The very high posterior standard deviation of the distance from the camera (t_Z) reflects the inability of the model to determine this parameter using only few landmark points (Table 5). The yaw angle does not show such a drastic performance difference and can be determined with much higher certainty from only few landmarks.

As a conclusion we can state that the orthographic camera model would suffice for frontal views when only few landmark points are available. The additional perspective of a pinhole camera cannot be resolved.

7.5 Face Recognition

We investigate the quality of our reconstructions in a recognition experiment. The reconstructed model instance provides a numerical representation of the face and can thus be used to compare two faces with a suitable similarity measure. To perform pose-invariant face recognition, we include only the model parameters $f = (\theta_S, \theta_C)$ in the similarity measure. We make use of the measure proposed in Blanz and Vetter (2003) with the 3DMM. The similarity s between two faces f_1 and f_2 in the model space is then

$$s = \frac{\langle f_1, f_2 \rangle}{\| f_1 \| \cdot \| f_2 \|}. \tag{26}$$

BFM recognition A small face recognition experiment was performed on the synthetic data provided with the BFM (Paysan et al. 2009). The set consists of renderings for 10 subjects over 9 poses and 3 illumination settings. The recognition setting was implemented as proposed by Aldrian and Smith (2013). The landmarks provided with the data were used to compare the algorithms. The respective galleries consist of a set of images with the same pose and illumination. Every image is used once as a probe and compared to all images over all possible combinations of galleries. The similarity is measured using (26). Our results are in the same range as results obtained by the method of Romdhani and Vetter (2005) and

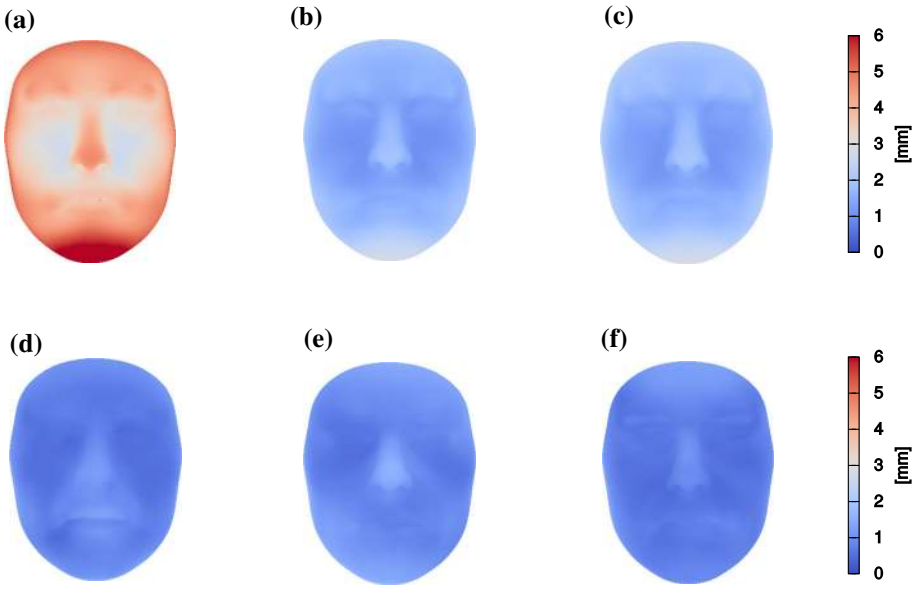

Fig. 13 Variability of face shape (standard deviation at given surface point) of the prior distribution (**a**) and different posteriors (**b**–**f**). Landmarks information constrains the shape where there is information available, both detected (**b**) and user-provided (**c**). The additional uncertainty in detection maps over user-provided landmarks is mainly absorbed in higher pose variability (see Table 6). Image information leads to a strong restriction of shape variability (**d**). The combined posteriors (**e**, **f**), conditioned on both the image and feature point cannot add more certainty compared to (**d**). The relatively high variability of the nose is very apparent in all posteriors. The exact reconstruction of the nose depth from projected frontal views is inherently ambiguous. **a** Prior, **b** detection, **c** landmarks, **d** image, **e** detection and image, **f** landmarks and image

Table 6 Posterior standard deviations of landmarks and image fits for a single example

Posterior	ϕ (yaw) (°)	t_Z (distance) (mm)
Detection	12	12,583
Landmarks	2.9	3,215
Image	0.5	19
Detection and image	0.5	48
Landmarks and image	0.4	28

Aldrian and Smith (2013) (all numbers from Aldrian and Smith 2013). Note that the given landmarks are very exact and therefore can be fully exploited by those optimization methods in a clean synthetic setting. Our method integrates them as unreliable and noisy input. The results are presented using shape and color coefficients separately in Tables 7 and 8.

Multi-PIE We conducted a large face recognition experiment on photographs using the Multi-PIE database (Gross et al. 2010). This database contains a systematic exploration of identity, pose, illumination and expression. We evaluate recognition performance on neutral photographs of the 249 individuals of the first session.

We use the frontal images from the first session as gallery and images at different yaw angles as probes. We then retrieve the most similar face from the gallery using (26) and calculate correct rank-1 identification rates. The exact identification of image sets is given in Table 9. This is the setup as proposed in Schönborn et al. (2013). Note that we did not adapt any part of our generic face reconstruction method to the Multi-PIE database. The only assumption we make is that there is exactly one face per image.

The proposed integrative inference method is able to achieve a fully automatic face reconstruction by using detection certainty maps as introduced above. We compare the recognition performance of the fully automatic method to a standard initialization with user-provided landmark positions which are fully reliable. Additionally, we compare with the naive feed-forward initialization using the single best detection results as certain inputs for initialization. The presented method of integrating feature point detection is currently limited to roughly 45° of yaw angle since it does not yet handle occlusion of feature points.

Detection For both face and facial feature detection, we use a standard random forest algorithm close to (Breiman 2001) with a scanning window to find the face. We grow a face detection forest and 9 feature point forests, all built in a very similar fashion using appropriate training patches. See Fig. 4 for a display of used landmarks.

A forest consists of 256 Haar-like feature trees with a maximal depth of 30. We use the information gain criterion

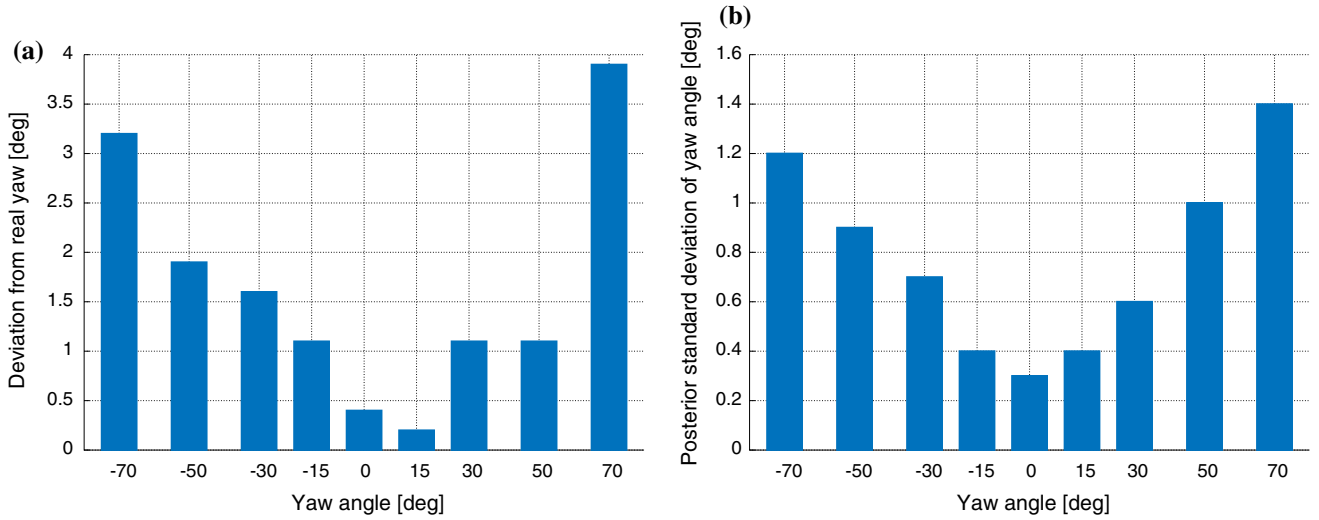

Fig. 14 Yaw estimation on BFM renderings (Paysan et al. 2009), deviation from ground truth (**a**) and estimation of the posterior standard deviation (**b**)

Table 7 Mean rank-1 recognition rates using the shape coefficients for the fittings of all 270 BFM renderings averaged over tree illumination conditions per pose

Pose	Aldrian	Romdhani	Ours
−70°	96.7	87.8	97.8
−50°	100	93.6	100
−30°	100	94.4	97.8
−15°	100	91.6	100
0 °	97.8	92.9	93.3
15°	98.9	90.7	100
30°	100	94.5	98.9
50°	100	96.3	98.9
70°	92.2	93.0	96.7
Mean	98.4	92.7	98.1

Table 8 Mean rank-1 recognition rates using the color coefficients for the fittings of all 270 BFM renderings averaged over tree illumination conditions per pose

Pose	Aldrian	Romdhani	Ours
−70°	92.0	81.0	78.9
−50°	94.8	92.0	84.9
−30°	94.9	89.9	90.5
−15°	98.4	91.7	94.1
0 °	95.9	91.0	88.6
15°	94.9	88.1	86.2
30°	94.4	82.6	90.9
50°	96.0	84.7	88.9
70°	95.8	85.9	86.9
Mean	95.0	87.4	87.8

to select a split from a set of many random candidates at each node. Training data is obtained as proposed by Eckhardt et al. (2009) from the very rich Annotated Facial Landmarks in the Wild database (AFLW) (Köstinger et al. 2011). We build bags of 30% of the training data to train each tree. To obtain a certainty output, each leaf node stores its ratio of positives to negatives. To strengthen this detector, we additionally used negative examples from the PASCAL VOC 2012 (Everingham et al. 2009) marked as not containing any person. Face patches are mirrored horizontally to increase the amount of training data. In total, we use 25,000 positive and 100,000 negative examples.

The face detector produces the 10 highest-rated candidates with an overlap of less than 60% of the area for each input image. For each candidate face box, we run facial feature point detection in an area roughly 40% larger than the detected face box. Facial feature points are only searched around the scale of the face candidate.

Results The results in Table 9 show only a weak performance deterioration of the integrated method compared to the fully reliable user-provided initialization data. The method is thus able to successfully use the detection information. The detection quality is not good enough to simply use the strongest detection result as a certain initializer in the naive construction. It is thus necessary to deal with the uncertainty in the detection results to obtain good results.

7.6 In the Wild

The 3DMM is most useful when applied to real-world images. Therefore, we test the reconstruction performance in this category of images on the dataset *Labeled Faces in*

Table 9 Rank-1 recognition rates (percent) on the Multi-PIE database (Gross et al. 2010) across yaw angle, obtained by using frontal (051_16) images as gallery and the respective pose images as probes (exact camera and lighting condition indicated in second row)

	15° 140_16	30° 130_16	45° 080_16
Manual annotation	96.0	82.7	85.5
Best detection	94.4	69.9	49.4
Proposed integrative method	93.2	91.6	79.1

The integrative method reaches almost the performance of user-provided initialization while the detections are not reliable enough to be used directly

the Wild (LFW) (Huang et al. 2007). This dataset contains many posed but unconstrained photographs of celebrities. We present a visual evaluation of fitting quality on a few representative example images from this database. Because the BFM cannot handle expressions, we include only neutral images.

Our overview in Fig. 15 reveals a quite pleasing fitting quality on clean images. The model fails to deal with strong outliers, such as heavy beards and expressions, as expected. We did not tune the model to this database in any way. Therefore it is not restricted to the main central face and sometimes chooses to reconstruct faces in the background.

8 Discussion

We would like to highlight two aspects of the experimental evaluation. First, using the probabilistic sampling method, we can now estimate the certainty of a 3DMM fit after conditioning on input data. The evaluations of the posterior variability nicely demonstrate the use of this information as a diagnostic research tool for further investigation. Second, the stochastic nature of the complete algorithm, including all integrative parts, leads to high-quality automatic fitting. It is directly applicable, for example, in fully automatic, general-purpose face recognition, even without any adaptation to the specific database used.

Posterior uncertainty The reported posterior estimates reflect only model certainty with respect to the given input. They do not measure actual reconstruction accuracy. Model certainty is an optimal value which could ideally be reached by the model and the given input data. The actual reconstruction accuracy is usually lower since the model is not a perfect model of the input data.

8.1 Gradients

We consider the lack of gradient usage an advantage, since we believe realistic and detailed skin models tend to become more stochastic and thus cannot provide gradients. As an example, consider procedural textures or stochastic Perlin noise. Both are tools used in the computer graphics community to synthesize varying texture (Perlin 1985). Also, already with the 3DMM, the varying domain of evaluation due to self-occlusion is a serious problem for gradient evaluation (Schönborn et al. 2015). Gradients of complex models, such as the 3DMM fitted to an image, also tend to be valid only very locally, making it difficult to design an optimization algorithm which can deal with local optima. Stochastic sampling methods provide a clean and elegant solution for avoiding the problems systematically. By extending the framework to include Hamilton Monte Carlo moves (Duane et al. 1987), gradient information can be integrated into the sampling framework.

8.2 Sampling and Optimization

Compared to standard optimization algorithms, a random sampling procedure is inefficient because (a) it will reject many solutions but only after an expensive evaluation and (b) it will deliver redundant results in regions of high probability.

The advantages of the sampling approach are its robustness with respect to bad updates (proposals), its stochastic nature to avoid local optima and its probabilistic output. While a traditional optimization algorithm suffers severely from bad update steps, the MH algorithm can just ignore them. Detours and redundancy are useful for exploring the solution space, and the probabilistic result can characterize the posterior distribution with more than only its local maximum. However, compared to a pure optimization algorithm, a sampler will usually not produce the sample at the maximal value of the posterior.

Local optimization proposals The integration of deterministic moves or even local optimization steps is very simple on an algorithmic level. They can be added as proposals. The resulting algorithm is a strong fitting method which might profit very much from the additional efficient moves. But due to inaccurate transition corrections, the result is no longer strictly the exact posterior distribution. In the face fitting application, we do not consider this a problem because the "real" likelihood of the problem is not known and each choice is a compromise anyway. But if statistical correctness is mandatory, there are methods for a formally flawless integration of full local optimization steps into the MH algorithm, e.g. Multiple-Try MH (Liu et al. 2000).

Performance The method's performance in terms of speed is lower than that of optimization-only strategies. The stochastic sampler, as used in the experiments, runs in approximately 10 minutes on current consumer hardware, single-threaded.

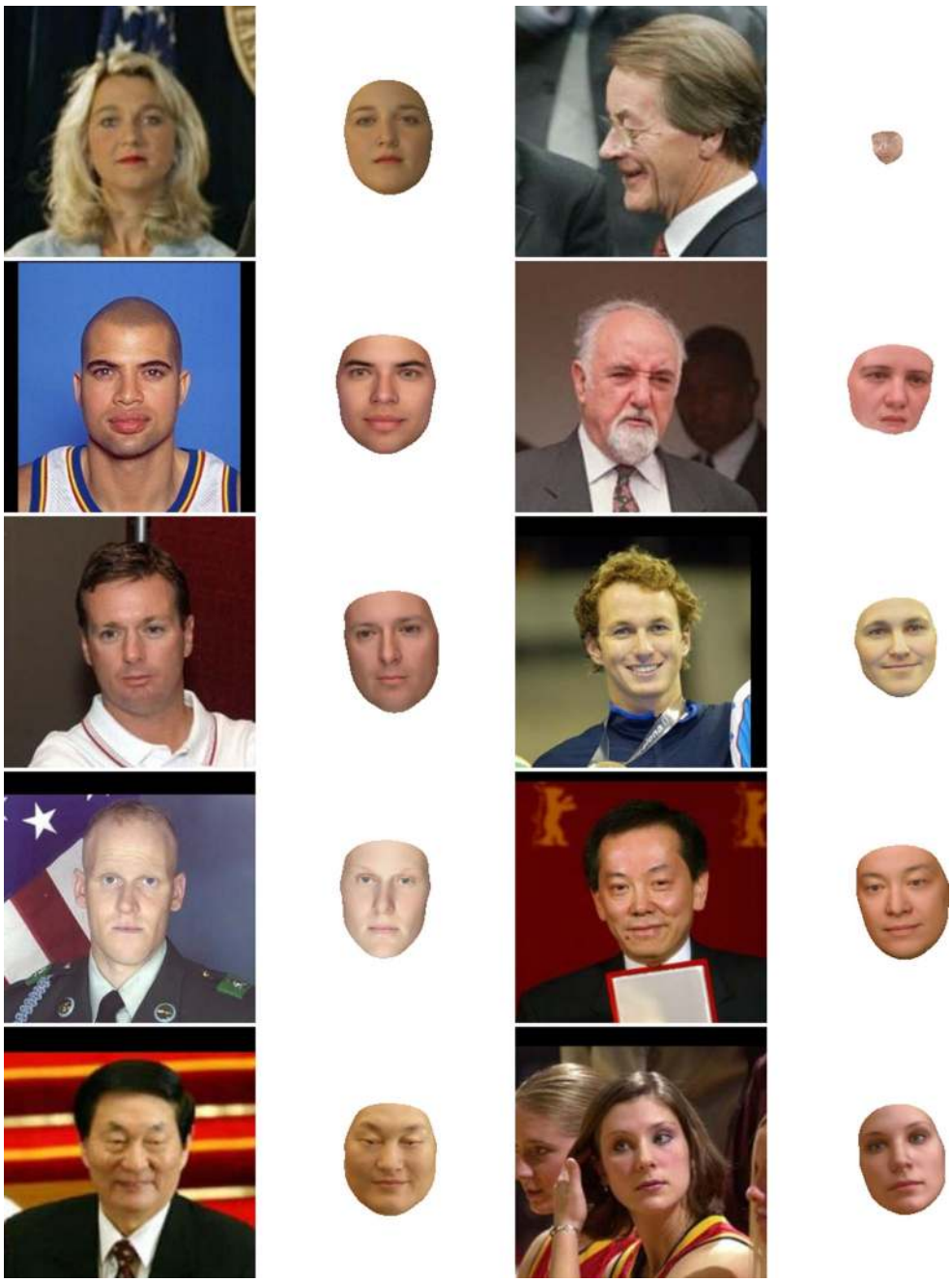

Fig. 15 This figure shows the performance of our method on the LFW database (Huang et al. 2007), which is close to a real world scenario. The *first* and *third column* show the original database images, the *second column* shows some successful fitting results of the fully automatic fitting process. The *fourth column* shows some frequent sources of errors (from *top to bottom*: detection failed, strong occlusion/beard, expression, textural details (mole), eye gaze)

This is still within the range of minutes that has always been the time necessary to adapt a full 3DMM. The long runtime is in large part due to the use of a software renderer and the high resolution of the face model. A further advantage of our method is a direct tradeoff of approximation quality with computation time. We can stop a sampling run at any time.

8.3 Comparison to Traditional Fitters

Our proposed method differs conceptually from traditional fitting methods, which all follow the optimization idea. Sampling aims at a fundamentally different result, a representation of the posterior distribution rather than a best-point estimate. However, a Markov Chain sampler, such as we pro-

Fig. 16 A case where a feed-forward initialization would fail. The most consistent face box at initialization is the *big yellow one*. The sampling avoids this early wrong decision. The *red facebox* leads in later sampling steps to higher consistency with all feature point detections and the image appearance. Therefore the sampling converges to the *red facebox*. A close up of the target image (*top*) and the final fitting result (*bottom*) are shown on the *right* (Image: KEYSTONE/EPA/Jason Szenes) (Color figure online)

pose, shows very similar behavior during a first burn-in phase when it searches for a region of high probability. Also the applications are quite similar and can be compared, especially if only the best sample is kept at the end. The single best sample of the sampler is not necessarily as well adapted as that of an optimization method.

A second feature of the proposed algorithm is the construction of a framework for integrating unreliable information directly into the inference process. The split into proposal and verification allows us to use unreliable proposals, as they can be rejected without disrupting the fitting process. Integration of this kind is difficult in traditional, gradient-based optimization methods. Consequently, we can set up the sampling fitter to act fully automatically without relying on high-quality initialization information.

In cases where the input information for initialization is available with high quality and reliability and only the single best solution is needed, a traditional optimizer can solve the problem considerably more efficiently. It will probably even reach a higher-quality output if the initialization is good enough.

When automatic methods are considered, the reliability of input information becomes an issue. We show an example of an image with a prominent face where the most consistent face (box and feature points) at initialization is the wrong one (Fig. 16). The proposed method can still recover and adapt the model to the face while a feed-forward initialization fails on this image.

We present a comparison of a few important properties of three state of the art fitting algorithms in Table 10.

The quantitative comparison shows a competitive outcome of the proposed method, even in synthetic reliable settings. But the more adapted method of Aldrian and Smith (2013) is very strong and efficient in settings where the feed-forward concept applies. We thus see the probabilistic sampling approach rather as a complementary method, applicable to different situations where more robustness or integration of different cues is necessary.

Stochastic gradient descent The stochastic gradient descent (SD) algorithm, proposed to adapt the original 3DMM in Blanz and Vetter (1999) appears to be somewhat similar to our method. Both methods are inherently stochastic and are thus not very prone to local optima. Apart from this similarity, there are many more differences. Albeit stochastic, stochastic gradient descent is not a probabilistic method. There is no information about the posterior distribution apart from its maximal value, as with any other optimization method. The stochastic nature of the algorithm arises from a partial evaluation of the gradient which leads to uncontrollable, adhoc randomness, as opposed to a well-defined proposal distribution in the MCMC fitter. Further, the propose-and-verify architecture of our method combined with filtering leads to a robust and integrative framework whereas stochastic gradient descent behaves like ordinary optimization in this respect and needs a proper initialization. Also, stochastic gradient descent still needs gradients, although only approximately.

Table 10 Comparison of different state of the art fitting algorithms

Method	Ours 2016	Romdhani 2005	Blanz 2003	Aldrian 2013
Model Type	PPCA	PCA	PCA	PPCA
Algorithm	DDMCMC	Levenberg-Marquardt (multi-stage)	Stochastic Newton	Bilinear, specific
Output	$P(\theta \mid \tilde{I})$	$\theta^\star = \arg\max_\theta P(\theta \mid \tilde{I})$	$\theta^\star$	$\theta^\star$
Input	Image, detections or landmarks	Image, landmarks, contour	Image, landmarks	Landmarks, image
Initialization	Automatic or manual	Landmarks (manual)	Manual	Landmarks (manual)
Camera	Perspective	Perspective	Perspective	Orthographic
Illumination	Spherical harmonics	Phong	Phong	Spherical harmonics with specular highlights
Evaluation	Image domain	Model (triangle centers)	Model (triangle centers)	Model (not important, geometry fixed for color evaluation)
Runtime	10 min	1 min	4 min	<second
Special	Probabilistic, robust, integrative	Multi-features, complicated multi-stage architecture	Stochastic	Linearization, sequential: geometry from landmarks only

We compare the most important methods to adapt a 3DMM to a single face image with respect to different categories and characteristics. Since no algorithm is published with code, the runtime is as reported in the respective publication. $\theta^\star$ refers to the locally optimal solution

9 Conclusion

We propose to solve the problem of fitting a parametric face model with Bayesian inference and do a full sampling-based approximation of the posterior distribution $P(\theta \mid \tilde{I})$. By using the MH algorithm, we not only get samples from the posterior distribution of the model's parameters given the input image but also a propose-and-verify concept. The separation into proposal and verification allows us to include uncertain und unreliable information directly into the inference process. By cascading multiple Metropolis acceptance steps, we successively integrate information of various origins, such as face and feature point detection or image pixel information, in a flexible, step-by-step manner. Together with the face and feature point detection presented, we construct a method for robust and fully automatic face reconstruction which does not rely on a single good initialization in a feed-forward manner. It can explore multiple hypotheses without a strong commitment to a single one determined in an uninformed initialization.

Inferring the posterior distribution, rather than just optimizing it, gives valuable insights into the certainty of a model fit. It can for example reveal the difficulty of finding the distance to the camera in a setup with only a few feature points or find the remaining variability of the face shape for a given target image, even for a single image. The stochastic nature of the algorithm avoids problems of local optima and provides robustness with respect to spurious false detections.

The evaluation of the method revealed a good performance in tasks such as 3D reconstruction of face shape and pose estimation from single images. The algorithm also enriches the application side through access to the posterior distribution which is useful for fitting diagnostics.

The downside, compared to traditional optimization-based fitters, is a poor efficiency and long runtime in situations where the flexibility and robustness of the presented framework are not necessary.

We believe this to be only a first variant of such an inference framework for face model fitting. Due to the propose-and-verify mechanism, the system is open to host even heuristic methods which are traditionally not used due to their unreliability. The model validation step decouples the updates from the actual model. This gives the user more freedom to design both, more complicated models and clever adaptation steps, which might be incomplete on their own. The current success of general DDMCMC applications in computer vision, sometimes termed *probabilistic programming*, is very promising in this respect.

Appendix 1: The Face Model

Face model The matrix **U** contains the principal components. The diagonal matrix **D** is modified slightly compared

to a standard PCA where it would contain the eigenvalues λ_i of the covariance matrix $\Sigma = \mathbf{U}\tilde{\mathbf{D}}^2\mathbf{U}^T$. We modify $\mathbf{D}$ to correspond with the proposed Maximum-Likelihood estimators from Tipping and Bishop (1999)

$$\mathbf{D}^2 = \tilde{\mathbf{D}}^2 - \sigma^2\mathbf{I} \tag{27}$$

We estimate the missing standard deviation parameter σ of the PPCA model as the Root Mean Square (RMS) reconstruction error of 3D faces, for both shape and texture. Reconstructing the 10 BFM out-of-sample faces, we obtain RMS reconstruction errors of $\hat{\sigma}_S = 0.61$ mm for the shape part and $\hat{\sigma}_C = 0.047$ for the color. Note that all color values are RGB floating point numbers in the interval [0, 1].

In order to better adapt the model to real images, we adapt only the face, without ears and throat. A rendering of the mean face of the masked model can be found in Fig. 2. We recalculate the statistics using only the restricted face mask to keep the model statistically valid and orthogonal.

All model parts of our software concerning statistical shape modeling are implemented using the *Statismo* framework (Lüthi et al. 2012).

Scene model The pinhole camera is modeled with focal length f and an offset $\mathbf{o}$ of the principal point within the image plane of size $w \times h$ pixels. The complete 3D-to-2D projection is then

$$\mathbf{x}_{2D} = \mathcal{P} \circ T \circ R_Z \circ R_Y \circ R_X \circ (\mathbf{x}_{3D}) \tag{28}$$

$$\mathcal{P}(\mathbf{r}) = \begin{bmatrix} wfr_x/r_z + o_x \\ hfr_y/r_z + o_y \end{bmatrix}. \tag{29}$$

Illumination The radiance p_i of a point i on the face surface with normal $\mathbf{n}_i$ and albedo a_i can be expressed using an expansion into real Spherical Harmonics basis functions Y_{lm}

$$p_i = a_i \sum_{l=0}^{2} \sum_{m=-l}^{l} Y_{lm}(\mathbf{n}_i) L_{lm} k_l. \tag{30}$$

The above equation is per color channel. The expansion of the environment map is captured in the illumination parameters L_{lm}, whereas the expansion of the Lambert reflectance kernel is given by k_l. For details, including the coefficient values k_l of the expansion, refer to Basri and Jacobs (2003).

The final image is produced by rasterization of all triangles in the face model. We use a Phong shading approach with a varying, interpolated normal for each pixel.

Illumination estimation As the light model (30) is linear for a given geometry, the illumination expansion coefficients L^c_{lm} for each color channel c are estimated solving a linear system (least squares) with entries for each vertex i

$$\sum_{l'=1}^{9} Y_{l'}(\mathbf{n}_i) k_{l'} a_i^c L_{l'}^c = p_i^c. \tag{31}$$

We solve the above system on 1000 randomly selected visible vertices i.

Product likelihood normalization The distribution is centered at the color value of the synthetic image and normalized to account for the truncation due to limited intensity values through

$$N = \int_0^1 \exp\left(-\frac{\|t - I_i(\theta)\|^2}{2\sigma^2}\right) dt_R dt_G dt_B \tag{32}$$

which can be calculated using the error function. The normalization can also be replaced by a standard Gaussian normalization as an approximation if the standard deviation is much smaller than the range of bounds and the color channels are neither saturated nor zero. This is a valid assumption for the majority of typical face images.

Appendix 2: Random Walk Proposals

The standard random walk proposal type is a Gaussian update:

$$Q: \quad \theta' = \theta + d \qquad d \sim \mathcal{N}(d|0, \sigma). \tag{33}$$

Camera model The proposals change three Euler angles of rotation, three directions of translation, the principal point in the image plane and the focal length. All of these are updated independently, only one at a time, using a selected variance for each. Additionally, 3D rotation proposals are compensated for unwanted movements of the face within the image plane such that it is kept at a fixed position in the image.

Face model The updates of the 3DMM's shape and texture models consist of two types of parameter variations. First, there is the addition of uncorrelated Gaussian noise to all parameters. Second, there is a scaling of the total parameter vector length with a log-normal distribution (distance from mean, "caricature"). Proposals are generated by

Table 11 Random walk proposals: σ is the standard deviation of the normal distribution, centered at the current location. λ designates mixture coefficients of the different scales coarse (C), intermediate (I) and fine (F). The values are obtained empirically

Parameter	Mixture					
	σ_C	σ_I	σ_F	λ_C	λ_I	λ_F
Yaw (rad)	0.75	0.1	0.01	0.1	0.4	0.5
Nick (rad)	0.75	0.1	0.01	0.1	0.4	0.5
Roll (rad)	0.75	0.1	0.01	0.1	0.4	0.5
Focal length, log f	0.15	0.05	0.01	0.2	0.6	0.2
Distance, t_z (mm)	500	50	5	0.2	0.6	0.2
Translation, $t_{x,y}$ (mm)	300	50	10	0.2	0.2	0.6
Shape, $\mathbf{q}_S$	0.2	0.1	0.025	0.1	0.5	0.2
Radial shape, $\|\mathbf{q}_S\|$	0.2			0.2		
Color, $\mathbf{q}_C$	0.2	0.1	0.025	0.1	0.5	0.2
Radial color, $\|\mathbf{q}_C\|$	0.2			0.2		
Light perturbation	0.001			1		
Light intensity, log f	0.1			1		
Light color	0.01			1		

$$Q_S: \quad \boldsymbol{\theta}'_{\mathrm{S}} = \boldsymbol{\theta}_{\mathrm{S}} + d \qquad d \sim \mathcal{N}(d|0, \sigma_{\mathrm{S}}\mathbf{I}) \tag{34}$$

$$\boldsymbol{\theta}'_{\mathrm{S}} = \boldsymbol{\theta}_{\mathrm{S}} \times \lambda \qquad \lambda \sim \log \mathcal{N}(1, \sigma_{\mathrm{SL}}). \tag{35}$$

Ilumination The illumination coefficients are updated with a mixture of a perturbation, an intensity and a color proposal. The perturbation is a standard independent Gaussian acting on all coefficients at once. The intensity proposal scales all coefficients by a factor drawn from a log-normal distribution and the color proposal keeps the intensity constant while perturbing the coefficients.

Table 11 contains a detailed overview.

References

Albrecht, T., Lüthi, M., Gerig, T., & Vetter, T. (2013). Posterior shape models. *Medical Image Analysis*, *17*(8), 959–973. doi:10.1016/j.media.2013.05.010.

Aldrian, O., & Smith, W. (2013). Inverse rendering of faces with a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(5), 1080–1093. doi:10.1109/TPAMI.2012.206.

Basri, R., & Jacobs, D. W. (2003). Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(2), 218–233.

Blanz, V., & Vetter, T. (1999). A morphable model for the synthesis of 3d faces. In SIGGRAPH '99: Proceedings of the 26th annual conference on computer graphics and interactive techniques (pp. 187–194). New York: ACM Press/Addison-Wesley. doi:10.1145/311535.311556.

Blanz, V., & Vetter, T. (2003). Face recognition based on fitting a 3D morphable model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(9), 1063–1074. doi:10.1109/TPAMI.2003.1227983.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. doi:10.1023/A:1010933404324.

Chib, S., & Greenberg, E. (1995). Understanding the Metropolis–Hastings algorithm. *The American Statistician*, *49*(4), 327–335.

Cootes, T., Edwards, G., & Taylor, C. (2001). Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *23*(6), 681–685. doi:10.1109/34.927467.

Duane, S., Kennedy, A. D., Pendleton, B. J., & Roweth, D. (1987). Hybrid Monte Carlo. *Physics Letters B*, *195*(2), 216–222.

Eckhardt, M., Fasel, I., & Movellan, J. (2009). Towards practical facial feature detection. *International Journal of Pattern Recognition and Artificial Intelligence*, *23*(03), 379–400.

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J., & Zisserman, A. (2009). The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, *88*(2), 303–338. doi:10.1007/s11263-009-0275-4.

Felzenszwalb, P. F., & Huttenlocher, D. P. (2012). Distance transforms of sampled functions. *Theory of Computing*, *8*(1), 415–428. doi:10.4086/toc.2012.v008a019.

Gilks, W. R., Richardson, S., & Spiegelhalter, D. J. (1996). *Markov chain Monte Carlo in practice* (Vol. 2). Boca Raton, FL: CRC Press.

Gonick, L., & Smith, W. (1993). *Cartoon guide to statistics*. New York: HarperCollins.

Gross, R., Matthews, I., Cohn, J., Kanade, T., & Baker, S. (2010). Multi-PIE. *Image and Vision Computing*, *28*(5), 807–813. doi:10.1016/j.imavis.2009.08.002.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, *57*(1), 97–109. doi:10.1093/biomet/57.1.97.

Huang, G.B., Ramesh, M., Berg, T., & Learned-Miller, E. (2007). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. Rep. 07-49, University of Massachusetts, Amherst.

Jampani, V., Nowozin, S., Loper, M., & Gehler, P. V. (2015). The informed sampler: A discriminative approach to bayesian inference in generative computer vision models. *Computer Vision and Image Understanding*, *136*, 32–44. doi:10.1016/j.cviu.2015.03.002.

Kirby, M., & Sirovich, L. (1990). Application of the Karhunen–Loeve procedure for the characterization of human faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *12*(1), 103–108.

Köstinger, M., Wohlhart, P., Roth, P. M., & Bischof, H. (2011). Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *2011 IEEE international conference on computer vision workshops (ICCV workshops)* (pp. 2144–2151).

Kulkarni, T. D., Kohli, P., Tenenbaum, J. B., & Mansinghka, V. (2015). Picture: A probabilistic programming language for scene perception. In *The IEEE conference on computer vision and pattern recognition (CVPR)*.

Liu, C., Shum, H. Y., & Zhang, C. (2002). Hierarchical shape modeling for automatic face localization. In *Computer Vision—ECCV 2002* (pp. 687–703). Heidelberg: Springer.

Liu, J. S., Liang, F., & Wong, W. H. (2000). The multiple-try method and local optimization in Metropolis sampling. *Journal of the American Statistical Association*, *95*(449), 121–134.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.

Lüthi, M., Blanc, R., Albrecht, T., Gass, T., Goksel, O., Buchler, P., et al. (2012). Statismo—A framework for PCA based statistical models. *The Insight Journal, 1*, 1–18.

Matthews, I., & Baker, S. (2004). Active appearance models revisited. *International Journal of Computer Vision*, *60*(2), 135–164.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, *21*, 1087.

Paysan, P., Knothe, R., Amberg, B., Romdhani, S., & Vetter, T. (2009). A 3D face model for pose and illumination invariant face recognition. In *Advanced video and signal based surveillance, 2009* (pp. 296–301).

Perlin, K. (1985). An image synthesizer. *ACM SIGGRAPH Computer Graphics*, *19*(3), 287–296.

Rauschert, I., & Collins, R. T. (2012). A generative model for simultaneous estimation of human body shape and pixel-level segmentation. In *Computer Vision—ECCV 2012* (pp. 704–717). Heidelberg: Springer.

Robert, C. P., & Casella, G. (2004). *Monte Carlo statistical methods* (Vol. 319). Citeseer.

Romdhani, S., & Vetter, T. (2003). Efficient, robust and accurate fitting of a 3D morphable model. In *Proceedings of ninth IEEE international conference on computer vision, 2003* (pp. 59–66).

Romdhani, S., & Vetter, T. (2005). Estimating 3D shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior. In *IEEE Computer Society conference on computer vision and pattern recognition, 2005 (CVPR 2005)* (Vol. 2, pp. 986–993). doi:10.1109/CVPR.2005.145.

Sambridge, M., & Mosegaard, K. (2002). Monte Carlo methods in geophysical inverse problems. *Reviews of Geophysics*, *40*(3), 1009. doi:10.1029/2000RG000089.

Schönborn, S., Egger, B., Forster, A., & Vetter, T. (2015). Background modeling for generative image models. *Computer Vision and Image Understanding*, *136*, 117–127. doi:10.1016/j.cviu.2015.01.008.

Schönborn, S., Forster, A., Egger, B., & Vetter, T. (2013). A Monte Carlo strategy to integrate detection and model-based face analysis. In J. Weickert, M. Hein, & B. Schiele (Eds.), *Pattern recognition*. Lecture notes in computer science (Vol. 8142, pp. 101–110). Berlin: Springer.

Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *61*(3), 611–622.

Tu, Z., Chen, X., Yuille, A. L., & Zhu, S. C. (2005). Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision*, *63*(2), 113–140. doi:10.1007/s11263-005-6642-x.

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, *3*(1), 71–86.

Wojek, C., Roth, S., Schindler, K., & Schiele, B. (2010). Monocular 3d scene modeling and inference: Understanding multi-object traffic scenes. In *Computer Vision—ECCV 2010* (pp. 467–481). Heidelberg: Springer.

Xiong, X., & De La Torre, F. (2013). Supervised descent method and its applications to face alignment. In *2013 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 532–539). doi:10.1109/CVPR.2013.75.

Yin, L., Wei, X., Sun, Y., Wang, J., & Rosato, M. (2006). A 3D facial expression database for facial behavior research. In *7th International conference on automatic face and gesture recognition, 2006 (FGR 2006)* (pp. 211–216). doi:10.1109/FGR.2006.6.

Zhu, X., Yan, J., Yi, D., Lei, Z., & Li, S. (2015). Discriminative 3d morphable model fitting. In *Proceedings of 11th IEEE international conference on automatic face and gesture recognition FG2015*, Ljubljana, Slovenia.

Zivanov, J., Forster, A., Schönborn, S., & Vetter, T. (2013). Human face shape analysis under spherical harmonics illumination considering self occlusion. In *6th International conference on biometrics, ICB-2013*, Madrid.