3D Reconstruction of "In-the-Wild" Faces in Images and Videos

James Booth, Anastasios Roussos, Evangelos Ververas, Epameinondas Antonakos, Stylianos Ploumpis, Yannis Panagakis, and Stefanos Zafeiriou

Abstract—3D Morphable Models (3DMMs) are powerful statistical models of 3D facial shape and texture, and among the state-of-the-art methods for reconstructing facial shape from single images. With the advent of new 3D sensors, many 3D facial datasets have been collected containing both neutral as well as expressive faces. However, all datasets are captured under controlled conditions. Thus, even though powerful 3D facial shape models can be learnt from such data, it is difficult to build statistical texture models that are sufficient to reconstruct faces captured in unconstrained conditions ("in-the-wild"). In this paper, we propose the first "in-the-wild" 3DMM by combining a statistical model of facial identity and expression shape with an "in-the-wild" texture model. We show that such an approach allows for the development of a greatly simplified fitting procedure for images and videos, as there is no need to optimise with regards to the illumination parameters. We have collected three new databases that combine "in-the-wild" images and video with ground truth 3D facial geometry, the first of their kind, and report extensive quantitative evaluations using them that demonstrate our method is state-of-the-art.

Index Terms—3DMM, Morphable Model, RPCA, 3D reconstruction.

1 Introduction

During the past few years, we have witnessed significant improvements in various face analysis tasks such as face detection [1], [2] and 2D facial landmark localisation on static images [3], [4], [5], [6], [7], [8], [9], [10]. This is primarily attributed to the fact that the community has made a considerable effort to collect and annotate facial images captured under unconstrained conditions [11], [12], [13], [14], [15] (commonly referred to as "in-the-wild") and to develop discriminative methodologies that can capitalise on the availability of such large amount of data. Nevertheless, discriminative techniques cannot be applied for 3D facial shape reconstruction "in-the-wild", due to lack of ground-truth data.

3D facial shape reconstruction from a single image or a video captured under "in-the-wild" conditions is still an open and challenging problem in Computer Vision. This is mainly due to the fact that the general problem of extracting the 3D facial shape from a single image, or even a video sequence, is an ill-posed problem which is notoriously difficult to solve without the use of any statistical priors for the shape and texture of faces. That is, without prior knowledge regarding the shape of the object at-hand there are inherent ambiguities present in the problem. The pixel intensity at a location in an image is the result of a complex combination of the underlying shape of the object, the surface albedo and normal characteristics, camera parameters and the arrangement of scene lighting and other objects in the scene. Hence, there are potentially infinite solutions to the problem.

Furthermore, learning statistical priors of the 3D facial shape and texture for "in-the-wild" images is currently very difficult by using modern acquisition devices. That is, even though there is a considerable improvement in 3D acquisition devices, they still cannot operate in arbitrary conditions. Hence, all the current 3D facial databases have been captured in controlled conditions.

With the available 3D facial data, it is feasible to learn a powerful statistical model of the facial shape that generalises well for both identity and expression [16], [17], [18]. However, it is not possible to construct a statistical model of the facial texture that generalises well for "in-the-wild" images and is, at the same time, in correspondence with the statistical shape model. That is the reason why current state-of-the-art 3D face reconstruction methodologies rely solely on fitting a statistical 3D facial shape prior on a sparse set of landmarks [19], [20].

In this paper, we make a number of contributions that enable the use of 3DMMs for "in-the-wild" face reconstruction (Fig. 1):

- Motivated by the success of feature-based (e.g., HOG [21], SIFT [22]) Active Appearance Models (AAMs) [8], [23], we propose a methodology for learning a statistical texture model from "in-the-wild" facial images, which is in full correspondence with a statistical shape prior that exhibits both identity and expression variations.
- By capitalising on the recent advancements in fitting statistical deformable models [8], [24], [25], [26], we propose a novel and fast algorithm for fitting our "in-the-wild" 3DMMs on images and videos. We show that the advantage of using the "in-the-wild" feature-based texture model is that the fitting strategy can be significantly simplified since there is no need to optimise with respect to illumination parameters.
- We make the implementation of our algorithm publicly available within the Menpo Project [27][1]. We strongly believe that this can be of great benefit to the community, given the lack of robust open-source implementations for fitting 3DMMs.

- *The authors are with the Department of Computing, Imperial College London, South Kensington Campus, London SW7 2AZ, UK.*
- *A. Roussos is also with the College of Engineering, Mathematics and Physical Sciences, University of Exeter, UK.*
E-mails: see https://ibug.doc.ic.ac.uk/people

1. https://github.com/menpo/itwmm

Fig. 1. Results of our 3DMM image fitting method *ITW(Basel)* on "in-the-wild" images from the 300W dataset [15]. We note that our proposed technique is able to handle extremely challenging pose, illumination, and expression variations, returning plausible 3D facial shapes in all the above cases.

- In order to provide quantitative evaluations we collect three new datasets which couple "in-the-wild" images with 3D ground truth shape information — KF-ITW, 4DMaja and 3dMD-Lab.
- We present extensive quantitative and qualitative evaluations of our proposed method against a wide range of state-of-the-art alternatives, which demonstrates the clear merits of our technique. [18].

The remainder of the paper is structured as follows. In Section 2 we briefly outline the background on face reconstruction from monocular cameras. In Section 3 we elaborate on the construction of our "in-the-wild" 3DMM, whilst in Section 4 we outline the proposed optimisation for fitting "in-the-wild" images with our model. Specifically, we extensively present our approach for fitting static images and videos in Sections 4.1 and 4.2, respectively. Section 5 describes our three new datasets, the first of their kind, which provide "in-the-wild" images and video sequences with ground-truth 3D facial shape. We outline a series of quantitative and qualitative experiments in Section 6, and end with conclusions in Section 7.

2 Background

Accurate recovery of the true 3D structure of a scene captured by an image or video is arguably one of the core problems in computer vision. Although it is feasible to recover many properties of a scene's background, the geometry of the objects within the scene is the most important task, since it enables the acquisition of powerful and descriptive models from which to perform inference. In particular, the 3D shape of the underlying objects is arguably the strongest cue for common tasks such as object recognition and localisation. However, the general problem or recovering the 3D shape of an object from a single image, or even a set of images with different viewpoints, is ill-conditioned. Even when provided with multiple images, additional information about the scene or details about the capturing conditions, 3D shape recovery is full of ambiguities. Many strategies have been proposed for solving this problem.

In contrast to the difficulty of the general case, the recovery of 3D facial shape has been successful in scenarios with controlled recording conditions. Human faces exhibit several characteristics that are beneficial for performing shape recovery: *(i)* they have approximately homogeneous configuration (all healthy human faces have the same parts, such as eyes, nose and mouth, in the same approximate locations), *(ii)* they have convex shape, and *(iii)* they exhibit approximately Lambertian reflectance [28], [29], [30], [31], [32], [33], [34], [35]. Nevertheless, the task is still very challenging since faces are highly deformable, their appearance changes dramatically depending on the illumination conditions and can exhibit severe self-occlusions depending on the viewpoint.

In this paper, we are interested in the very challenging problem of 3D face reconstruction from still images or videos captured under unconstrained conditions, i.e. "in-the-wild". Hence, we herein review methodologies that do not require the use of any specialised machinery (e.g., depth or stereo cameras).

Although the relevant literature is very extensive, a categorisation of sorts can be structured as follows:

Shape-from-Shading (SfS): These methods expect a single image [36] (or a collection of images) as input and use image formation assumptions (usually the Lambertian reflectance assumption) to recover surface shape. There is considerable research in SfS for generic surfaces, as well as faces [36], [37], [38], [39], [40], [41], [42]. However, generic SfS techniques do not produce very convincing results for faces [39], unless face shape priors are introduced [36], [38] or jointly performing SfS in a large collection of facial images [41], [42]. The current state-of-the-art techniques include methods such as [41], [42], which even though they are able to recover some facial details, they require dense alignment to be performed (e.g., by using elaborate optical flow techniques [42]) and they are only suitable for recovering 2.5D information and not full 3D shape.

3D Morphable Models (3DMM): The 3DMM fitting proposed in the work of Blanz & Vetter [43], [44] was among the first model-based 3D facial recovery approaches. The first 3DMM was built using 200 faces captured in well-controlled conditions displaying only the neutral expression. That is the reason why the method was only shown to work on real-world, but not "in-the-wild", images. Since then, many extensions have been proposed to the original method [45], [46], [47], [48]. Although model-based SfS may also consider similarity to a facial model as a measure of reconstruction accuracy, 3DMMs are unique in explicitly *rendering* images of faces for the purpose of 3D recovery. Until recently, due to the lack of available texture models, 3DMMs were deemed suitable only for images captured under controlled conditions. Hence, many works considered only fitting a dense shape model to a collection of sparse landmarks that were localised in the image [19], [20]. In this paper, we make a significant step further and demonstrate how to train the first in-the-wild 3DMM.

Structure-from-Motion (SfM): These methods employ geometric constraints in order to recover 3D structure across multiple images or frames of a sequence. Although the majority of research in this area is not face specific, facial data is commonly used to demonstrate the effectiveness of a method [49]. Nevertheless, the lack of use of appropriate facial shape models makes the problem of dense 3D face reconstruction very difficult to solve. This is due to the fact that the dense SfM requires the solution of a very high dimensional non-convex optimisation problem [49] which also assumes the presence of very accurate dense flow [50], something that makes such techniques applicable mainly in controlled recording conditions [49]. Nevertheless, sparse SfM applied on a collection of tracked landmarks can be used to provide an initialisation to our methodology when it comes to reconstructing faces in videos.

3 Model Training

A 3DMM consists of three parametric models: the *shape*, *camera* and *texture* models.

3.1 Shape Modelling

Let us denote the 3D mesh (shape) of an object with N vertices as a $3N \times 1$ vector

$$\mathbf{s} = \left[\mathbf{x}_1^\mathsf{T}, \ldots, \mathbf{x}_N^\mathsf{T}\right]^\mathsf{T} = [x_1, y_1, z_1, \ldots, x_N, y_N, z_N]^\mathsf{T} \quad (1)$$

where $\mathbf{x}_i = [x_i, y_i, z_i]^\mathsf{T}$ are the object-centered Cartesian coordinates of the i-th vertex.

We first of all consider an identity shape model, i.e. a model of shape variation across different individuals, assuming that all shapes are under neutral expression. For this, we adopt the recently released LSFM model [18], the largest-scale 3D Morphable Model (3DMM) of facial identity built from around 10,000 scans of different individuals.

A 3D shape model like the one in LSFM is constructed by first bringing a set of 3D training meshes into dense correspondence so that each is described with the same number of vertices and all samples have a shared semantic ordering. The corresponded meshes, $\{\mathbf{s}_i\}$, are then brought into a shape space by applying Generalised Procrustes Analysis and then Principal Component Analysis (PCA) is performed which results in $\{\bar{\mathbf{s}}_{id}, \mathbf{U}_{id}, \mathbf{\Sigma}_{id}\}$, where $\bar{\mathbf{s}}_{id} \in \mathbb{R}^{3N}$ is the mean shape vector, $\mathbf{U}_{id} \in \mathbb{R}^{3N \times n_p}$ is the orthonormal basis after keeping the first n_p principal components and $\mathbf{\Sigma}_{id} \in \mathbb{R}^{n_p \times n_p}$ is a diagonal matrix with the standard deviations of the corresponding principal components. Let $\widetilde{\mathbf{U}}_{id} = \mathbf{U}_{id}\mathbf{\Sigma}_{id}$ be the identity basis with basis vectors that have absorbed the standard deviation of the corresponding mode of variation so that the shape parameters $\mathbf{p} = [p_1, \ldots, p_{n_p}]^\mathsf{T}$ are normalised to have unit variance. Therefore, assuming normal prior distributions, we have $\mathbf{p} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n_p})$, where $\mathbf{I}_n$ denotes the $n \times n$ identity matrix. Also, a 3D shape instance of a novel identity can be generated using this model as a function of the parameters $\mathbf{p}$:

$$\mathcal{S}_{id}(\mathbf{p}) = \bar{\mathbf{s}}_{id} + \widetilde{\mathbf{U}}_{id}\mathbf{p} \quad (2)$$

Visualisations of the the identity model are included in the Supplementary Material.

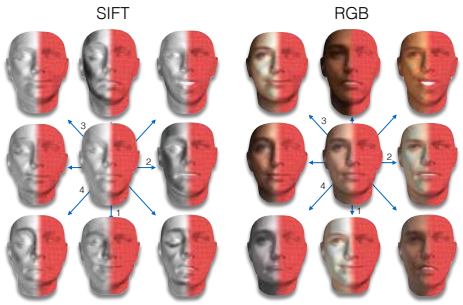


Fig. 2. *Left:* The mean and first four shape and SIFT texture principal components of our "in-the-wild" SIFT texture model. *Right:* To aid in interpretation we also show the equivalent RGB basis.

Furthermore, we also consider a 3D shape model of expression variations, as offsets from a given identity shape $\mathcal{S}_{id}$. For this we use the blendshapes model of Facewarehouse [16]. We adopt Nonrigid ICP [51] to accurately register this model with the LSFM identity model. After this procedure, the expression model can be represented with the triplet $\{\bar{\mathbf{s}}_{exp}, \mathbf{U}_{exp}, \mathbf{\Sigma}_{exp}\}$, where $\bar{\mathbf{s}}_{exp} \in \mathbb{R}^{3N}$ is the mean expression offset, $\mathbf{U}_{exp} \in \mathbb{R}^{3N \times n_q}$ is the orthonormal expression basis having n_q principal components and $\mathbf{\Sigma}_{exp} \in \mathbb{R}^{n_q \times n_q}$ is the diagonal matrix with the corresponding standard deviations. Similarly with the identity model, we consider the basis $\widetilde{\mathbf{U}}_{exp} = \mathbf{U}_{exp}\mathbf{\Sigma}_{exp}$ and the associated normalised parameters $\mathbf{q} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{n_q})$.

Combining the two aforementioned models, we end up with the following combined model that represents the 3D facial shape of any identity under any expression:

$$\mathcal{S}(\mathbf{p}, \mathbf{q}) = \bar{\mathbf{s}} + \widetilde{\mathbf{U}}_{id}\mathbf{p} + \widetilde{\mathbf{U}}_{exp}\mathbf{q} \quad (3)$$

where $\bar{\mathbf{s}} = \bar{\mathbf{s}}_{id} + \bar{\mathbf{s}}_{exp}$ is the overall mean shape, $\mathbf{p}$ is the vector with the identity parameters and $\mathbf{q}$ is the vector with the expression parameters.

3.2 Camera Model

The purpose of the camera model is to map (project) the object-centred Cartesian coordinates of a 3D mesh instance $\mathbf{s}$ into 2D Cartesian coordinates on an image plane.

The projection of a 3D point $\mathbf{x} = [x, y, z]^\mathsf{T}$ into its 2D location in the image plane $\mathbf{x}' = [x', y']^\mathsf{T}$ involves two steps. First, the 3D point is rotated and translated using a linear *view transformation* to bring it in the camera reference frame:

$$\mathbf{v} = [v_x, v_y, v_z]^\mathsf{T} = \mathbf{R}_v\mathbf{x} + \mathbf{t}_v \quad (4)$$

where $\mathbf{R}_v \in \mathbb{R}^{3\times3}$ and $\mathbf{t}_v = [t_x, t_y, t_z]^\mathsf{T}$ are the camera's 3D rotation and translation components, respectively. This is based on the fact that, without loss of generality, we can assume that the observed facial shape is still and that the relative change in 3D pose between camera and object is only due to camera motion.

Then, a camera projection is applied as:

$$\mathbf{x}' = \pi(\mathbf{c}_{\text{intr}}, \mathbf{v}) \tag{5}$$

where $\mathbf{c}_{\text{intr}}$ is a vector with the camera's intrinsic parameters.

The above generic formulation can be applied to any camera model. For example, in the case of a perspective camera with its principal point fixed at the image centre, $\mathbf{c}_{\text{intr}} = \phi$, where ϕ is the focal length and the camera projection function is defined as:

$$\pi(\mathbf{v}, \phi) = \frac{\phi}{v_z} \begin{bmatrix} v_x \\ v_y \end{bmatrix} + \begin{bmatrix} c_x \\ c_y \end{bmatrix} \tag{6}$$

where $[c_x, c_y]^\mathsf{T}$ are the image coordinates of the image centre.

In the case of a scaled orthographic camera projection, $\mathbf{c}_{\text{intr}} = \sigma$, where σ is the scale parameter of the camera and the camera projection function is given by:

$$\pi(\mathbf{v}, \sigma) = \sigma \begin{bmatrix} v_x \\ v_y \end{bmatrix} \tag{7}$$

Quaternions. We parametrise the 3D rotation with quaternions [52], [53]. The quaternion uses four parameters $\mathbf{q} = [q_0, q_1, q_2, q_3]^\mathsf{T}$ in order to express a 3D rotation as

$$\mathbf{R}_v = 2 \begin{bmatrix} \frac{1}{2} - q_2^2 - q_3^2 & q_1 q_2 - q_0 q_3 & q_1 q_3 + q_0 q_2 \\ q_1 q_2 + q_0 q_3 & \frac{1}{2} - q_1^2 - q_3^2 & q_2 q_3 - q_0 q_1 \\ q_1 q_3 - q_0 q_2 & q_2 q_3 + q_0 q_1 & \frac{1}{2} - q_1^2 - q_2^2 \end{bmatrix} \tag{8}$$

Note that by enforcing a unit norm constraint on the quaternion vector, i.e. $\mathbf{q}^\mathsf{T}\mathbf{q} = 1$, the rotation matrix constraints of orthogonality with unit determinant are withheld. Given the unit norm property, the quaternion can be seen as a three-parameter vector $[q_1, q_2, q_3]^\mathsf{T}$ and a scalar $q_0 = \sqrt{1 - q_1^2 - q_2^2 - q_3^2}$. Most existing works on 3DMM parametrise the rotation matrix $\mathbf{R}_v$ using the three Euler angles that define the rotations around the horizontal, vertical and camera axes. Even thought Euler angles are more naturally interpretable, they have strong disadvantages when employed within an optimisation procedure, most notably the solution ambiguity and the gimbal lock effect.

Camera function. The projection operation performed by the camera model of the 3DMM can be expressed with the function $\mathcal{P}(\mathbf{s}, \mathbf{c}) : \mathbb{R}^{3N} \rightarrow \mathbb{R}^{2N}$, which applies the transformations of Eqs. (4) and (6) on the points of provided 3D mesh $\mathbf{s}$ with

$$\mathbf{c} = [\mathbf{c}_{\text{intr}}, q_1, q_2, q_3, t_x, t_y, t_z]^\mathsf{T} \tag{9}$$

being the vector of *camera parameters* with length $n_c = 7$. For abbreviation purposes, we represent the camera model of the 3DMM with the function $\mathcal{W} : \mathbb{R}^{n_p, n_c} \rightarrow \mathbb{R}^{2N}$ as

$$\mathcal{W}(\mathbf{p}, \mathbf{q}, \mathbf{c}) \equiv \mathcal{P}\left(\mathcal{S}(\mathbf{p}, \mathbf{q}), \mathbf{c}\right) \tag{10}$$

where $\mathcal{S}(\mathbf{p}, \mathbf{q})$ is a 3D mesh instance using Eq. (2).

3.3 Feature-Based Texture Model

The generation of "in-the-wild" texture models is a key component of the proposed 3DMM. We build feature-based texture models by avoiding the estimation of illumination parameters. This leads to a more efficient and robust representation. To construct such models, it would not be effective to use the texture from 3D facial scans, as usually done in the construction of 3DMMs [18], [43], since the illumination conditions are excessively controlled in such scans. On contrary, our goal is to model the texture of faces, as captured by images and videos under completely uncontrolled conditions. Therefore, we utilise a large collection of in-the-wild facial images, accompanied with a sparse set of facial landmarks.

We assume that for the aforementioned set of M "in-the-wild" images $\{\mathbf{I}_i\}_1^M$, we have access to the associated camera and shape parameters $\{\mathbf{p_i}, \mathbf{q_i}, \mathbf{c_i}\}$. These parameters are initially estimated by fitting the combined 3D shape model on the sparse 2D landmarks. Let us also define a *dense* feature extraction function

$$\mathcal{F} : \mathbb{R}^{H\times W\times N_{\text{colors}}} \rightarrow \mathbb{R}^{H\times W\times C} \tag{11}$$

where H, W, N_{colors} are the width, height and number of color channels respectively of the input image and C is the number of channels of the feature-based image. For each image, we first compute its feature-based representation as $\mathbf{F}_i = \mathcal{F}(\mathbf{I}_i)$ and then use Eq. (10) to sample it at each vertex location to build back a vectorised texture sample $\mathbf{t}_i = \mathbf{F}_i\left(\mathcal{W}(\mathbf{p_i}, \mathbf{q_i}, \mathbf{c_i})\right) \in \mathbb{R}^{CN}$. This texture sample will be nonsensical for some regions mainly due to self-occlusions present in the mesh projected in the image space $\mathcal{W}(\mathbf{p_i}, \mathbf{q_i}, \mathbf{c_i})$. To alleviate these issues, we cast a ray from the camera to each vertex and test for self-intersections with the triangulation of the mesh in order to learn a per-vertex occlusion mask $\mathbf{m}_i \in \mathbb{R}^N$ for the projected sample.

Let us create the matrix $\mathbf{X} = [\mathbf{t}_1, \ldots, \mathbf{t}_M] \in \mathbb{R}^{CN\times M}$ by concatenating the M grossly corrupted feature-based texture vectors with missing entries that are represented by the masks $\mathbf{m}_i$. To robustly build a texture model based on this incomplete data, we need to recover a low-rank matrix $\mathbf{L} \in \mathbb{R}^{CN\times M}$ representing the clean facial texture and a sparse matrix $\mathbf{E} \in \mathbb{R}^{CN\times M}$ accounting for gross but sparse non-Gaussian noise such that $\mathbf{X} = \mathbf{L} + \mathbf{E}$. To simultaneously recover both $\mathbf{L}$ and $\mathbf{E}$ from incomplete and grossly corrupted observations, the Principal Component Pursuit with missing values [54] is solved

$$\begin{aligned} &\arg\min_{\mathbf{L},\mathbf{E}} \|\mathbf{L}\|_* + \lambda\|\mathbf{E}\|_1 \\ &\text{s.t. } \mathcal{P}_\Omega(\mathbf{X}) = \mathcal{P}_\Omega(\mathbf{L} + \mathbf{E}), \end{aligned} \tag{12}$$

where $\|\cdot\|_*$ denotes the nuclear norm, $\|\cdot\|_1$ is the matrix ℓ_1-norm and $\lambda > 0$ is a regularizer. Ω represents the set of locations corresponding to the observed entries of $\mathbf{X}$ (i.e., $(i,j) \in \Omega$ if $m_i = m_j = 1$). Then, $\mathcal{P}_\Omega(\mathbf{X})$ is defined as the projection of the matrix $\mathbf{X}$ on the observed entries Ω, namely $\mathcal{P}_\Omega(\mathbf{X})_{ij} = x_{ij}$ if $(i,j) \in \Omega$ and $\mathcal{P}_\Omega(\mathbf{X})_{ij} = 0$ otherwise. The unique solution of the convex optimization problem in Eq. (12) is found by employing an Alternating Direction Method of Multipliers-based algorithm [55].

The final texture model is created by applying PCA on $\mathbf{L}$ (the set of reconstructed feature-based textures acquired from the previous procedure). This results in $\{\bar{\mathbf{t}}, \mathbf{U}_t\}$, where $\bar{\mathbf{t}} \in \mathbb{R}^{CN}$ is the mean texture vector and $\mathbf{U}_t \in \mathbb{R}^{CN\times n_t}$ is the orthonormal basis after keeping the first n_t principal components. This model can be used to generate novel 3D feature-based texture instances with the function $\mathcal{T} : \mathbb{R}^{n_t} \rightarrow \mathbb{R}^{CN}$ as

$$\mathcal{T}(\boldsymbol{\lambda}) = \bar{\mathbf{t}} + \mathbf{U}_t\boldsymbol{\lambda} \tag{13}$$

where $\boldsymbol{\lambda} = [\lambda_1, \ldots, \lambda_{n_t}]^\mathsf{T}$ are the n_t *texture parameters*.

Finally, an iterative procedure is used in order to refine the texture. That is, we started with the 3D fits provided by using only

Fig. 3. Building an ITW texture model. The red coloured region denotes the occlusion mask obtained by fitting the 3D shape model on the sparse 2D landmarks of the original image.

the 2D landmarks [56]. Then, a texture model is learnt using the above procedure. The texture model was used with the proposed 3DMM fitting algorithm on the same data and texture model was refined. This could be repeated over multiple iterations, but we have empirically found that a single refinement iteration is adequate. In the case of single-image fitting, this procedure is done in a separate training phase, which needs to be performed only once. In the case of video fitting, this can be done for every input video.

4 Model Fitting

We propose an energy minimisation formulation to fit the 3DMM on single images and videos. We design an efficient optimisation strategy, which is based on the Gauss-Newton method and the Project-Out approach. To this end, herein, we first present the fitting on single images and then proceed with the fitting on videos.

4.1 Fitting on Single Images

4.1.1 Proposed Energy Formulation

To fit the 3DMM on single images, we propose to minimise the following cost function:

$$E(\mathbf{p},\mathbf{q},\mathbf{c},\boldsymbol{\lambda}) = E_{\text{text}}(\mathbf{p},\mathbf{q},\mathbf{c},\boldsymbol{\lambda}) + c_\ell E_{\text{land}}(\mathbf{p},\mathbf{q},\mathbf{c}) + E_{\text{priors}}(\mathbf{p},\mathbf{q}) \tag{14}$$

where E_{text} is a texture reconstruction term, E_{land} is a sparse 2D landmarks term and E_{priors} is a shape priors term that regularises the shape parameters. Also c_ℓ is the balancing weight of the E_{land} term. The energy E depends on the shape $(\mathbf{p},\mathbf{q})$, texture $\boldsymbol{\lambda}$ and camera $\mathbf{c}$ parameters and these are the quantities that we seek to estimate by minimising it. The terms E_{land} and E_{priors} are optional and aim to facilitate the optimisation procedure in order to converge faster and to a better minimum. Note that thanks to the proposed "in-the-wild" feature-based texture model, the cost function does not include any parametric illumination model similar to the ones in the related literature [43], [44], which greatly simplifies the optimisation. Next, we present every term of the energy.

The **texture reconstruction term (E_{text})** is the main data term of the optimisation problem. It depends on shape, texture and camera parameters and penalises the squared L^2 norm of the difference between the image feature-based texture that corresponds to the projected 2D locations of the 3D shape instance and the texture instance of the 3DMM:

$$E_{\text{text}}(\mathbf{p},\mathbf{q},\mathbf{c},\boldsymbol{\lambda}) = \left\|\mathbf{F}\left(\mathcal{W}(\mathbf{p},\mathbf{q},\mathbf{c})\right) - \mathcal{T}(\boldsymbol{\lambda})\right\|^2 \tag{15}$$

where $\mathbf{F} = \mathcal{F}(\mathbf{I})$ denotes the feature-based representation with C channels of an input image $\mathbf{I}$ using Eq. (11). Note that $\mathbf{F}\left(\mathcal{W}(\mathbf{p},\mathbf{q},\mathbf{c})\right) \in \mathbb{R}^{CN}$ denotes the operation of sampling the feature-based input image on the projected 2D locations of the 3D shape instance acquired by the camera model (Eq. (10)).

The **2D landmarks term (E_{land})** is an auxiliary data term that is based on sparse 2D landmarks:

$$E_{\text{land}}(\mathbf{p},\mathbf{q},\mathbf{c}) = \left\|\mathcal{W}_l(\mathbf{p},\mathbf{q},\mathbf{c}) - \boldsymbol{\ell}\right\|^2 \tag{16}$$

where $\boldsymbol{\ell} = [x_1, y_1, \ldots, x_L, y_L]^{\mathsf{T}}$ denotes a set of L sparse 2D landmark points ($L \ll N$) defined on the image coordinate system and $\mathcal{W}_l(\mathbf{p},\mathbf{q},\mathbf{c})$ returns the $2L \times 1$ vector of 2D projected locations of these L sparse landmarks. Intuitively, this term aims to drive the optimisation procedure using the selected sparse landmarks as anchors for which we have the optimal locations $\boldsymbol{\ell}$. In this way, the camera parameters can be rapidly adapted.

The **shape priors term (E_{priors})** aims at avoiding overfitting effects and penalizes reconstructed faces that are unlikely to happen, under the consider shape model. It consists of two optional prior terms over the identity and expression parameters, $\mathbf{p}$ and $\mathbf{q}$. Based on the normal distributions assumptions for $\mathbf{p}$ and $\mathbf{q}$ and the fact that these are normalised (see Sec. 3.1), we formulate the prior terms as the squared L^2 norms of the parameters:

$$E_{\text{priors}}(\mathbf{p},\mathbf{q}) = c_{id}\left\|\mathbf{p}\right\|^2 + c_{exp}\left\|\mathbf{q}\right\|^2 \tag{17}$$

where c_{id} and c_{exp} are constants that weight the contribution of the prior terms over identity and expression parameters respectively.

4.1.2 Gauss-Newton Project-Out Optimisation

Inspired by the extensive literature in Lucas-Kanade 2D image alignment [8], [24], [25], [26], [57], [58], we formulate a Gauss-Newton optimization framework to efficiently minimize the energy of Eq. (14).

Parameters update. The shape and camera parameters are updated in an additive manner, i.e.

$$\mathbf{p} \leftarrow \mathbf{p} + \Delta\mathbf{p}, \quad \mathbf{q} \leftarrow \mathbf{q} + \Delta\mathbf{q}, \quad \mathbf{c} \leftarrow \mathbf{c} + \Delta\mathbf{c} \tag{18}$$

where $\Delta\mathbf{p}$, $\Delta\mathbf{q}$ and $\Delta\mathbf{c}$ are their increments estimated at each fitting iteration. Note that in the case of the quaternion used to parameterize the 3D rotation matrix, the update is performed as the multiplication

$$\begin{aligned}\mathbf{q} \leftarrow (\Delta\mathbf{q})\mathbf{q} &= \begin{bmatrix} \Delta q_0 \\ \Delta\mathbf{q}_{1:3} \end{bmatrix} \begin{bmatrix} q_0 \\ \mathbf{q}_{1:3} \end{bmatrix} = \\ &= \begin{bmatrix} \Delta q_0 q_0 - \Delta\mathbf{q}_{1:3}^{\mathsf{T}}\mathbf{q}_{1:3} \\ \Delta q_0\mathbf{q}_{1:3} + q_0\Delta\mathbf{q}_{1:3} + \Delta\mathbf{q}_{1:3} \times \mathbf{q}_{1:3} \end{bmatrix}\end{aligned} \tag{19}$$

However, we will still denote it as an addition for simplicity. Finally, we found that it is beneficial to keep the focal length constant in most cases, due to its ambiguity with t_z.

Linearisation. By introducing the additive incremental updates on the shape and camera parameters, the cost function is expressed as:

$$\begin{aligned}&E(\mathbf{p}+\Delta\mathbf{p},\mathbf{q}+\Delta\mathbf{q},\mathbf{c}+\Delta\mathbf{c},\boldsymbol{\lambda}) = \\ &\quad\left\|\mathbf{F}\left(\mathcal{W}(\mathbf{p}+\Delta\mathbf{p},\mathbf{q}+\Delta\mathbf{q},\mathbf{c}+\Delta\mathbf{c})\right) - \mathcal{T}(\boldsymbol{\lambda})\right\|^2 \\ &\quad + c_\ell\left\|\mathcal{W}_l(\mathbf{p}+\Delta\mathbf{p},\mathbf{q}+\Delta\mathbf{q},\mathbf{c}+\Delta\mathbf{c}) - \boldsymbol{\ell}\right\|^2 \\ &\quad + c_{id}\left\|\mathbf{p}+\Delta\mathbf{p}\right\|^2 + c_{exp}\left\|\mathbf{q}+\Delta\mathbf{q}\right\|^2\end{aligned} \tag{20}$$

Note that the texture reconstruction and landmarks constraint terms of this cost function are non-linear due to the camera model

operation. We need to linearise them around $(\mathbf{p}, \mathbf{q}, \mathbf{c})$ using first order Taylor series expansion at $(\mathbf{p}+\Delta\mathbf{p}, \mathbf{q}+\Delta\mathbf{q}, \mathbf{c}+\Delta\mathbf{c}) = (\mathbf{p}, \mathbf{q}, \mathbf{c}) \Rightarrow (\Delta\mathbf{p}, \Delta\mathbf{q}, \Delta\mathbf{c}) = \mathbf{0}$. The linearisation for the image term gives:

$$\begin{aligned}\mathbf{F}\left(\mathcal{W}(\mathbf{p}+\Delta\mathbf{p}, \mathbf{q}+\Delta\mathbf{q}, \mathbf{c}+\Delta\mathbf{c})\right) \approx \mathbf{F}\left(\mathcal{W}(\mathbf{p}, \mathbf{q}, \mathbf{c})\right) \\ + \mathbf{J}_{\mathbf{F},\mathbf{p}}\Delta\mathbf{p} + \mathbf{J}_{\mathbf{F},\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{\mathbf{F},\mathbf{c}}\Delta\mathbf{c}\end{aligned} \tag{21}$$

where:

$$\mathbf{J}_{\mathbf{F},\mathbf{p}} = \nabla\mathbf{F}\ \frac{\partial\mathcal{W}}{\partial\mathbf{p}}\Big|_{\mathbf{p}=\mathbf{p}}, \mathbf{J}_{\mathbf{F},\mathbf{q}} = \nabla\mathbf{F}\ \frac{\partial\mathcal{W}}{\partial\mathbf{q}}\Big|_{\mathbf{q}=\mathbf{q}}, \mathbf{J}_{\mathbf{F},\mathbf{c}} = \nabla\mathbf{F}\ \frac{\partial\mathcal{W}}{\partial\mathbf{c}}\Big|_{\mathbf{c}=\mathbf{c}}$$

are the *image Jacobians* with respect to the identity, expression and camera parameters, respectively. Note that most dense feature-extraction functions $\mathcal{F}(\cdot)$ are non-differentiable, thus we simply compute the gradient of the multi-channel feature image $\nabla\mathbf{F}$.

Similarly, the linearisation on the sparse landmarks projection term gives:

$$\begin{aligned}\mathcal{W}_l(\mathbf{p}+\Delta\mathbf{p}, \mathbf{q}+\Delta\mathbf{q}, \mathbf{c}+\Delta\mathbf{c}) \approx \\ \mathcal{W}_l(\mathbf{p}, \mathbf{q}, \mathbf{c}) + \mathbf{J}_{L,\mathbf{p}}\Delta\mathbf{p} + \mathbf{J}_{L,\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{L,\mathbf{c}}\Delta\mathbf{c}\end{aligned} \tag{22}$$

where: $\mathbf{J}_{L,\mathbf{p}} = \frac{\partial\mathcal{W}_l}{\partial\mathbf{p}}\Big|_{\mathbf{p}=\mathbf{p}}, \mathbf{J}_{L,\mathbf{q}} = \frac{\partial\mathcal{W}_l}{\partial\mathbf{q}}\Big|_{\mathbf{q}=\mathbf{q}}, \mathbf{J}_{L,\mathbf{c}} = \frac{\partial\mathcal{W}_l}{\partial\mathbf{c}}\Big|_{\mathbf{c}=\mathbf{c}}$

are the *landmarks projection Jacobians*. Please refer to the supplementary material for more details on the computation of these derivatives.

By substituting Eqs. (21) and (22) into Eq. (20) the cost function is approximated as:

$$\begin{aligned}&E(\mathbf{p}+\Delta\mathbf{p}, \mathbf{q}+\Delta\mathbf{q}, \mathbf{c}+\Delta\mathbf{c}, \boldsymbol{\lambda}) \approx \\ &\left\|\mathbf{F}\left(\mathcal{W}(\mathbf{p}, \mathbf{q}, \mathbf{c})\right) + \mathbf{J}_{\mathbf{F},\mathbf{p}}\Delta\mathbf{p} + \mathbf{J}_{\mathbf{F},\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{\mathbf{F},\mathbf{c}}\Delta\mathbf{c} - \mathcal{T}(\boldsymbol{\lambda})\right\|^2 \\ &+ c_\ell \left\|\mathcal{W}_l(\mathbf{p}, \mathbf{q}, \mathbf{c}) + \mathbf{J}_{L,\mathbf{p}}\Delta\mathbf{p} + \mathbf{J}_{L,\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{L,\mathbf{c}}\Delta\mathbf{c} - \boldsymbol{\ell}\right\|^2 \\ &+ c_{id}\left\|\mathbf{p}+\Delta\mathbf{p}\right\|^2 + c_{exp}\left\|\mathbf{q}+\Delta\mathbf{q}\right\|^2\end{aligned} \tag{23}$$

Adopting the **Project-Out** optimisation approach, we optimise on the orthogonal complement of the texture subspace which eliminates the need to consider a texture parameters increment at each iteration. In more detail, the minimisation of the energy of Eq. (23) with respect to $\boldsymbol{\lambda}$ can be expressed analytically as a function of the increments $\Delta\mathbf{p}, \Delta\mathbf{q}, \Delta\mathbf{c}$:

$$\begin{aligned}\boldsymbol{\lambda} = \mathbf{U}_t{}^{\mathsf{T}}\Big(&\mathbf{F}(\mathcal{W}(\mathbf{p}, \mathbf{q}, \mathbf{c})) + \mathbf{J}_{\mathbf{F},\mathbf{p}}\Delta\mathbf{p} \\ &+ \mathbf{J}_{\mathbf{F},\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{\mathbf{F},\mathbf{c}}\Delta\mathbf{c} - \bar{\mathbf{t}}\Big)\end{aligned} \tag{24}$$

We plug this expression into Eq. (23) to eliminate the dependence of the energy on $\boldsymbol{\lambda}$ and we get the following minimisation problem:

$$\begin{aligned}&\underset{\Delta\mathbf{p},\Delta\mathbf{q},\Delta\mathbf{c}}{\arg\min} \\ &\left\|\mathbf{F}\left(\mathcal{W}(\mathbf{p}, \mathbf{q}, \mathbf{c})\right) + \mathbf{J}_{\mathbf{F},\mathbf{p}}\Delta\mathbf{p} + \mathbf{J}_{\mathbf{F},\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{\mathbf{F},\mathbf{c}}\Delta\mathbf{c} - \bar{\mathbf{t}}\right\|_{\mathbf{P}}^2 \\ &+ c_\ell \left\|\mathcal{W}_l(\mathbf{p}, \mathbf{q}, \mathbf{c}) + \mathbf{J}_{L,\mathbf{p}}\Delta\mathbf{p} + \mathbf{J}_{L,\mathbf{q}}\Delta\mathbf{q} + \mathbf{J}_{L,\mathbf{c}}\Delta\mathbf{c} - \boldsymbol{\ell}\right\|^2 \\ &+ c_{id}\left\|\mathbf{p}+\Delta\mathbf{p}\right\|^2 + c_{exp}\left\|\mathbf{q}+\Delta\mathbf{q}\right\|^2\end{aligned} \tag{25}$$

where $\mathbf{P} = \mathbf{I}_{CN} - \mathbf{U}_t\mathbf{U}_t{}^{\mathsf{T}}$ is the orthogonal complement of the texture subspace that functions as the "project-out" operator. Note that in this formulation $\boldsymbol{\lambda}$ plays no explicit role. Further note that in order to derive Eq. (25), we use the properties $\mathbf{P}^{\mathsf{T}} = \mathbf{P}$ and $\mathbf{P}^{\mathsf{T}}\mathbf{P} = \mathbf{P}$.

The problem of Eq. (25) is a linear least squares problem that can be written in the general compact form:

$$\underset{\Delta\mathbf{b}}{\arg\min}\left\|\mathbf{J}\Delta\mathbf{b} - \mathbf{e}\right\|^2 \tag{26}$$

where $\Delta\mathbf{b} = [\Delta\mathbf{p}^{\mathsf{T}}, \Delta\mathbf{q}^{\mathsf{T}}, \Delta\mathbf{c}^{\mathsf{T}}]^{\mathsf{T}}$ is a vector with all the unknowns (incremental updates) and $\mathbf{J}$ is the overall Jacobian of the problem:

$$\mathbf{J} = \left[\mathbf{J}_{\mathbf{p}} \,\middle|\, \mathbf{J}_{\mathbf{q}} \,\middle|\, \mathbf{J}_{\mathbf{c}}\right] = \left[\begin{array}{c|c|c} \mathbf{P}\mathbf{J}_{\mathbf{F},\mathbf{p}} & \mathbf{P}\mathbf{J}_{\mathbf{F},\mathbf{q}} & \mathbf{P}\mathbf{J}_{\mathbf{F},\mathbf{c}} \\ \sqrt{c_\ell}\,\mathbf{J}_{L,\mathbf{p}} & \sqrt{c_\ell}\,\mathbf{J}_{L,\mathbf{q}} & \sqrt{c_\ell}\,\mathbf{J}_{L,\mathbf{c}} \\ \sqrt{c_{id}}\,\mathbf{I}_{n_p} & \mathbf{0}_{n_p\times n_q} & \mathbf{0}_{n_p\times n_c} \\ \mathbf{0}_{n_q\times n_p} & \sqrt{c_{exp}}\,\mathbf{I}_{n_q} & \mathbf{0}_{n_q\times n_c} \end{array}\right] \tag{27}$$

where $\mathbf{0}_{m\times n}$ denotes the $m\times n$ zero matrix. Also, $\mathbf{e}$ is the overall offset vector of the problem:

$$\mathbf{e} = \left[\begin{array}{c} \mathbf{P}\left(\bar{\mathbf{t}} - \mathbf{F}\left(\mathcal{W}(\mathbf{p}, \mathbf{q}, \mathbf{c})\right)\right) \\ \sqrt{c_\ell}\left(\boldsymbol{\ell} - \mathcal{W}_l(\mathbf{p}, \mathbf{q}, \mathbf{c})\right) \\ -\sqrt{c_{id}}\,\mathbf{p} \\ -\sqrt{c_{exp}}\,\mathbf{q} \end{array}\right] \tag{28}$$

We compute $\Delta\mathbf{b}$ by solving the linear system that is derived from taking the gradient of the cost function in Eq. (26) and setting it to zero: $(\mathbf{J}^{\mathsf{T}}\mathbf{J})\Delta\mathbf{b} = \mathbf{J}^{\mathsf{T}}\mathbf{e}$. This system is of a relatively small scale, therefore it is straight-forward to implement its solution.

Note that the above-described Project-Out scheme is a very efficient approach to solving the Gauss-Newton iterations for minimising the cost function of Eq. (14). It has been shown that this is much faster than the more widely-used *Simultaneous algorithm* [23], [25], [59].

Residual masking. In practice, we apply a mask on the texture reconstruction residual of the Gauss-Newton optimisation, in order to speed-up the 3DMM fitting. This mask is constructed by first acquiring the set of visible vertices using z-buffering and then randomly selecting K of them. By keeping the number of vertices small ($K \approx 5000 \ll N$), we manage to greatly speed-up the fitting process without any accuracy penalty. This z-buffering and random sampling is performed per-iteration, allowing for changes in the self-occlusion state of vertices as the optimisation progresses.

4.2 Fitting on Videos

In the case of videos, we extend our energy minimisation formulation, described in the previous Section 4.1. Due to our separable identity and expression shape model, we can fix the identity parameters throughout the whole video, a significant constraint that greatly helps our estimations. In addition, we impose temporal smoothness on the expression parameters, which improves the estimation of the 3D facial deformations of the individual observed in the input video. Furthermore, we can get a fast and accurate initialisation for the minimisation of the proposed energy by employing Structure from Motion on the per-frame sparse 2D landmarks with an efficient linear least squares fitting approach.

4.2.1 Proposed Energy Formulation

Let us assume that the input video consists of n_f images, $\mathbf{I}_1, \ldots, \mathbf{I}_f, \ldots, \mathbf{I}_{n_f}$. As in the single-image case, we are based on the feature-based representation $\mathbf{F}_f = \mathcal{F}(\mathbf{I}_f)$ of the image of every frame $f = 1, \ldots, n_f$. Also, let $\boldsymbol{\ell}_f = [x_{1f}, y_{1f}, \ldots, x_{Lf}, y_{Lf}]^{\mathsf{T}}$ be the 2D facial landmarks for the f-th frame. We are still denoting by $\mathbf{p}$ the identity parameters

vector, which as already mentioned, is fixed over all frames of the video. However, we consider that every frame has its own expression, camera, and texture parameters vectors, which we denote by $\mathbf{q}_f$, $\mathbf{c}_f$ and $\boldsymbol{\lambda}_f$ respectively. We also denote by $\hat{\mathbf{q}}$, $\hat{\mathbf{c}}$ and $\hat{\boldsymbol{\lambda}}$ the concatenation of the corresponding parameter vectors over all frames: $\hat{\mathbf{q}}^\mathsf{T} = \left[\mathbf{q}_1^\mathsf{T}, \dots, \mathbf{q}_{n_f}^\mathsf{T}\right]$, $\hat{\mathbf{c}}^\mathsf{T} = \left[\mathbf{c}_1^\mathsf{T}, \dots, \mathbf{c}_{n_f}^\mathsf{T}\right]$ and $\hat{\boldsymbol{\lambda}}^\mathsf{T} = \left[\boldsymbol{\lambda}_1^\mathsf{T}, \dots, \boldsymbol{\lambda}_{n_f}^\mathsf{T}\right]$

To fit the 3DMM on a video, we propose to minimise the following energy, which is a multi-frame extension of the energy in Eq. (14):

$$\begin{aligned}\hat{E}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}, \hat{\boldsymbol{\lambda}}) = \hat{E}_{\text{text}}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}, \hat{\boldsymbol{\lambda}}) + c_\ell \hat{E}_{\text{land}}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}) \\ + \hat{E}_{\text{priors}}(\mathbf{p}, \hat{\mathbf{q}}) + c_{sm} \hat{E}_{\text{smooth}}(\hat{\mathbf{q}})\end{aligned} \tag{29}$$

where $\hat{E}_{\text{text}}$, $\hat{E}_{\text{land}}$ and $\hat{E}_{\text{priors}}$ are the multi-frame extensions of the texture reconstruction, 2D landmarks term and prior regularisation terms respectively. Furthermore, $\hat{E}_{\text{smooth}}$ is a temporal smoothness term that we impose on the time-varying expression parameters $\mathbf{q}_f$. Also c_ℓ and c_{sm} are the balancing weights for the terms $\hat{E}_{\text{land}}$ and $\hat{E}_{\text{smooth}}$ respectively. Next, we present every term of the energy in more detail.

The **texture reconstruction term ($\hat{E}_{\text{text}}$)** is the main data term and sums the texture reconstruction error from all frames:

$$\hat{E}_{\text{text}}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}, \hat{\boldsymbol{\lambda}}) = \sum_{f=1}^{n_f} \left\| \mathbf{F}_f \left(\mathcal{W}(\mathbf{p}, \mathbf{q}_f, \mathbf{c}_f) \right) - \mathcal{T}(\boldsymbol{\lambda}_f) \right\|^2 \tag{30}$$

The **2D landmarks term ($\hat{E}_{\text{land}}$)** is a summation of the reprojection error of the sparse 2D landmarks for all frames:

$$\hat{E}_{\text{land}}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}) = \sum_{f=1}^{n_f} \left\| \mathcal{W}_l(\mathbf{p}, \mathbf{q}_f, \mathbf{c}_f) - \boldsymbol{\ell}_f \right\|^2 \tag{31}$$

The **shape priors term ($\hat{E}_{\text{priors}}$)** imposes priors on the reconstructed 3D facial shape of every frame. Since the facial shape at every frame is derived from the (zero-mean and unit-variance) identity parameter vector $\mathbf{p}$ and the frame-specific expression parameter vector $\mathbf{q}_f$ (also zero-mean and unit-variance), we define this term as:

$$\begin{aligned}\hat{E}_{\text{priors}}(\mathbf{p}, \hat{\mathbf{q}}) &= \hat{c}_{id} \left\| \mathbf{p} \right\|^2 + c_{exp} \sum_{f=1}^{n_f} \left\| \mathbf{q}_f \right\|^2 \\ &= \hat{c}_{id} \left\| \mathbf{p} \right\|^2 + c_{exp} \left\| \hat{\mathbf{q}} \right\|^2\end{aligned} \tag{32}$$

where $\hat{c}_{id}$ and c_{exp} are the balancing weights for the prior terms of identity and expression respectively.

The **temporal smoothness term ($\hat{E}_{\text{smooth}}$)** is video-specific and enforces smoothness on the expression parameters vector $\mathbf{q}_f$ by penalising the squared norm of the discrimination of its 2nd temporal derivative. This corresponds to the regularisation imposed in smoothing splines and leads to naturally smooth trajectories over time. More specifically, this term is defined as:

$$\hat{E}_{\text{smooth}}(\hat{\mathbf{q}}) = \sum_{f=2}^{n_f-1} \left\| \mathbf{q}_{f-1} - 2\mathbf{q}_f + \mathbf{q}_{f+1} \right\|^2 = \left\| \mathbf{D}^2 \hat{\mathbf{q}} \right\|^2 \tag{33}$$

where the summation is done over all frames for which the discretised 2nd derivative can be expressed without having to assume any form of padding outside the temporal window of the video. Also $\mathbf{D}^2 : \mathbb{R}^{n_q n_f} \rightarrow \mathbb{R}^{n_q (n_f - 2)}$ is the linear operator that instantiates the discretised 2nd derivative of the n_q-dimensional vector $\mathbf{q}_f$. This means that $\mathbf{D}^2 \hat{\mathbf{q}}$ is a vector that stacks the vectors $(\mathbf{q}_{f-1} - 2\mathbf{q}_f + \mathbf{q}_{f+1})$, for f=2, …, $n_f - 1$. It is worth mentioning that we could have imposed temporal smoothness on the parameters $\mathbf{c}_f$, $\boldsymbol{\lambda}_f$ too. However, we have empirically observed that the temporal smoothness on $\mathbf{q}_f$, in conjunction with fixing the identity parameters $\mathbf{p}$ over time, is adequate for accurate and temporally smooth estimations.

4.2.2 Initialisation

The proposed energy $\hat{E}$ in Eq. (29) is highly non-convex, therefore a good initialisation is of paramount importance. To achieve highly-accurate fitting results on videos, even in especially challenging cases, we design a computationally efficient video initialisation strategy, by decomposing the problem into two simpler ones that can be solved quickly and accurately.

For the above reasons, we consider for this part a scaled orthographic camera, which simplifies the optimisation by making the projection function $\pi(\mathbf{c}_{\text{intr}}, \mathbf{v})$ described in Eq. (6) to be linear with respect to $\mathbf{v}$. Also, we are based on a simplified version of the proposed energy $\hat{E}$ in Eq. (29) that does not contain the texture reconstruction term:

$$\begin{aligned}\hat{E}_{\text{init}}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}) = c_\ell \hat{E}_{\text{land}}(\mathbf{p}, \hat{\mathbf{q}}, \hat{\mathbf{c}}) \\ + \hat{E}_{\text{priors}}(\mathbf{p}, \hat{\mathbf{q}}) + c_{sm} \hat{E}_{\text{smooth}}(\hat{\mathbf{q}})\end{aligned} \tag{34}$$

This means that the only data term is $\hat{E}_{\text{land}}$ and the estimations use only the sparse 2D landmarks as input. Full details are provided in the Supplimentary Material.

4.2.3 Video-Specific Texture Model

Apart from offering a good starting point for the main optimisation, the initialisation described in the previous sections is first of all used to bootstrap the learning of the video-specific texture model, as described in Sec. 3.3. To improve the computational efficiency of this procedure, we down-sample the frames and only consider 1 every f_{step} frames. In more detail, using the estimated shape and camera parameters of the considered frames, we sample the facial texture $\mathbf{t}_f = \mathbf{F}_f \left(\mathcal{W}(\mathbf{p}, \mathbf{q_f}, \mathbf{c_f}) \right)$ and utilise it in the Principal Component Pursuit (PCP) problem of Eq. (12).

4.2.4 Main Optimisation of the Proposed Energy

Similarly to the single-image case (Sec. 4.1.2), we minimise the proposed energy $\hat{E}$ of Eq. (29) by following a Gauss-Newton scheme. In every iteration, we consider the current estimates $\mathbf{p}$, $\hat{\mathbf{q}}$, $\hat{\mathbf{c}}$ and we linearise the texture reconstruction and landmarks error functions around them. After this approximation, the problem becomes a linear least squares problem with respect to the texture parameters $\hat{\boldsymbol{\lambda}}$ and the incremental updates $\Delta\mathbf{p}$, $\Delta\hat{\mathbf{q}}$ and $\Delta\hat{\mathbf{c}}$. For more details, please see Supplementary Material.

Regarding the unknown texture parameters, we follow again the Project-Out approach. In more detail, the minimisation with respect to each $\boldsymbol{\lambda}_f$ is decoupled in every frame and can be found analytically as a function of $\Delta\mathbf{p}$, $\Delta\hat{\mathbf{q}}_f$ and $\Delta\hat{\mathbf{c}}_f$, exactly as in Eq. (24) (see Supplementary Material). Using this expression in the expression of the linearised energy $\hat{E}$, we derive the following problem:

$$\underset{\Delta\mathbf{p}, \Delta\hat{\mathbf{q}}, \Delta\hat{\mathbf{c}}}{\arg\min} \; \hat{E}\left(\mathbf{p} + \Delta\mathbf{p}, \hat{\mathbf{q}} + \Delta\hat{\mathbf{q}}, \hat{\mathbf{c}} + \Delta\hat{\mathbf{c}}\right) \tag{35}$$

The above problem is a large-scale linear least squares problem that can be written in the form (see Supplementary Material for detailed derivations):

$$\arg\min_{\Delta\hat{\mathbf{b}}} \|\hat{\mathbf{J}}\Delta\hat{\mathbf{b}} - \hat{\mathbf{e}}\|^2 \tag{36}$$

where $\Delta\hat{\mathbf{b}} = [\Delta\mathbf{p}^\mathsf{T}, \Delta\hat{\mathbf{q}}^\mathsf{T}, \Delta\hat{\mathbf{c}}^\mathsf{T}]$ is a vector with all the unknown incremental updates from all the frames. Also, $\hat{\mathbf{J}}$ is the corresponding overall Jacobian matrix that has a sparse structure. Finally, $\hat{\mathbf{e}}$ is the overall error term. Note that the dimensionality of $\Delta\hat{\mathbf{b}}$ (and hence the number of parameters to estimate) is $N_{\text{tot}} = n_p + n_f(n_q + 7)$ and the Jacobian $\hat{\mathbf{J}}$ is of size $(n_f(CN + 2L + n_q + 1) - 2n_q) \times N_{\text{tot}}$. Given the fact that we consider hundreds of frames n_f and tens of thousands of vertices N, the least square problem (36) is a very large-scale one. For example, for the choice of parameters considered in our experiments, the Jacobian $\hat{\mathbf{J}}$ is of size $425{,}884{,}944\times35{,}100$. This is in contrast to the corresponding problem of the single-image fitting case, where the problem was of small scale, so we could solve it by standard approaches. Therefore, we follow a video-specific strategy, in order to achieve a satisfactory scalability. In more detail, we consider the equivalent linear system (derived by equating the gradient to zero): $\hat{\mathbf{J}}^\mathsf{T}\hat{\mathbf{J}}\mathbf{x} = \hat{\mathbf{J}}^\mathsf{T}\mathbf{b}$ and adopt an efficient and parallelisable method that avoids the explicit computation and storage of the matrices $\hat{\mathbf{J}}$ and $(\hat{\mathbf{J}}^\mathsf{T}\hat{\mathbf{J}})$, which are very large-scale and sparse. More precisely, following other recent methods of 3D facial and more general deformable surface reconstruction [48], [60], we use a *preconditioned conjugate gradient* (PCG) solver, for which we only need to efficiently implement functions that compute the multiplications $(\hat{\mathbf{J}}^\mathsf{T}\hat{\mathbf{J}})\mathbf{x}$ and $\hat{\mathbf{J}}^\mathsf{T}\mathbf{h}$ for any input vectors $\mathbf{x}$ and $\mathbf{h}$. For the preconditioning, we use the inverses of the diagonal blocks of $\hat{\mathbf{J}}^\mathsf{T}\hat{\mathbf{J}}$.

5 Benchmark Datasets for 3DMM Image and Video Fitting

To allow for the quantitative evaluation of our proposed 3DMM image and video fitting methods, we have constructed three datasets — KF-ITW, 3dMD-Lab and 4DMaja. For the benefit of the research community, we are making these benchmark datasets publicly available[2]. We now describe each dataset in turn.

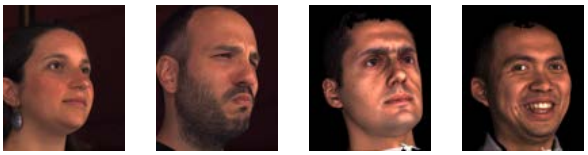
5.1 KF-ITW Dataset

The first dataset we introduce is focused on providing quantitative evaluation for 3DMM image fitting. KF-ITW is, to the best of our knowledge, the first dataset where ground truth 3D facial shape is provided along with images captured under relatively unconstrained conditions.

The dataset consists of 17 different subjects captured under various illumination conditions performing a range of expressions (*neutral*, *happy*, *surprise*). We employed the KinectFusion [61], [62] framework to acquire a 3D representation of the subjects with a Kinect v1 sensor. In order to accurately reconstruct the entire surface of the face, each subject was instructed to stay still in a fixed pose whilst a circular motion scanning pattern was carried out around the face. The fused mesh for each subject recovered from KinectFusion serves as a 3D face ground-truth in which we can evaluate our algorithm and compare it to other methods. Single frames picked from the RGB video stream of the Kinect sensor recording are selected as input images of our benchmark. The frame rate for every subject was constant to 8 frames per second, and a voxel grid of size 608^3 was utilised to get the detailed 3D scans of the faces. After getting the 3D scans from the KinectFusion framework we manually annotate each mesh with the iBUG 49 sparse landmark set, and use these landmarks to constrain a fit of the mean of the shape model in a non-rigid manner to each raw scan by performing a Non-Rigid Iterative Closest Point (N-ICP) matching between the two surfaces. This results in a set of meshes in dense correspondence, that is to say that all meshes in KF-ITW have the same well-behaved mesh topology and number and distribution of vertices for the evaluation process.

Although a short video sequence is captured as part of the acquisition process, we do not consider KF-ITW a suitable dataset for video facial shape recovery evaluation, as this video is highly contrived (the user is requested to hold still the entire time and the video motion is very specific and unnatural). To this end we only supply single frames from the acquisition process for image-fitting evaluation.

The evaluation protocol for KF-ITW is as follows. We use the ground-truth annotations provided in the KF-ITW dataset to initialize and fit each technique under test to the "in-the-wild" style images in the dataset. Our error metric is the per-vertex dense error between the recovered shape and the model-specific corresponded ground-truth fit, normalized by the inter-ocular distance for the test mesh (i.e. the distance between the outer corners of the eyes). Only regions of the face that are recovered by all methods under test can be used for evaluation. In our case, this corresponds to the inner region of the face. The neck, ears, and other extremities are not considered, as they only appear in a subset of models used by methods under test.



(a) (b) (c) (d)

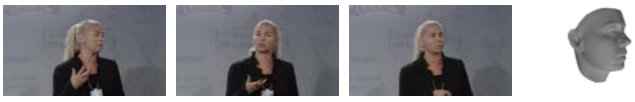
Fig. 4. 3dMD-Lab benchmark: (a,b) Examples of 2 out of 8 images of 3dMD-Lab(real images). We introduce this benchmark to evaluate image fitting methods under ideal conditions. (c,d) Examples of 2 out of 8 images of 3dMD-Lab(synthetic images). We introduce this benchmark to evaluate image fitting methods under synthetic strong illumination conditions.

Fig. 5. 4DMaja(synthetic video) benchmark: 4 out of 440 frames of a synthetic video created using high-resolution 4D face scans and rendering using a synthetic camera under varying 3D pose. Since this is a rendered video, it is accompanied by 4D ground truth mesh information.

5.2 3dMD-Lab Dataset

To quantitatively evaluate additional aspects of 3DMM image fitting, we are introducing a second benchmark dataset, which we call "3dMD-Lab". In contrast to KF-ITW, this dataset has been

2. webpage:https://goo.gl/2DwhHz



(a) (b) (c) (d)

Fig. 6. 4DMaja(real video) benchmark: (a-c) 3 out of 387 frames of a real video under in-the-wild conditions. (d) Ground truth mesh representing the shape identity component of the 3D facial shape of the captured subject.

created in more controlled conditions and has been based on high-resolution 3D face scans, using a $3dMD^{TM}$ face scanner. This makes the ground truth facial meshes to be highly-detailed, and allows us to to evaluate the performance of 3DMM image fitting methods in the use case where we have no "in-the-wild" effects.

In more detail, 3dMD-Lab includes 4 subjects each performing 2 different expressions for a total of eight 3D face scans. It includes 8 real images ("3dMD-Lab(real images)") in ideal, laboratory conditions, coming directly from one of the RGB cameras of the $3dMD^{TM}$ face scanning system. It also includes 8 synthetic images ("3dMD-Lab(synthetic images)") created by the same scans after rendering them from different view points and with added illumination directed light under different directions. All real and synthetic images are high-resolution images with spatial dimension of 2048×2448 pixels and true colour range (24 bits per pixel). Please see see Supplementary Material for more details on both these variations, including sample images.

5.3 4DMaja Dataset

To quantitatively evaluate 3DMM **video** fitting, we are introducing a third benchmark, which we call "4DMaja". To the best of our knowledge, this is the first publicly available benchmark that allows detailed quantitative evaluation of 3D face reconstruction on videos. 4DMaja includes two face videos of the same subject (Prof. Maja Pantic) under varying natural expressions and significant head pose variation. The first video is a synthetic video created based on high-resolution 4D face scans, using a $DI4D^{TM}$ face scanner: see Fig. 5. In more detail, the video was created by using a 4D scan of the subject under different expressions and rendering it with a synthetic camera that undergoes a periodic rotation. This allows the comparison over 4D ground truth information, i.e. quantitative evaluation of the 3D face reconstruction for every frame of the video. This video includes 440 frames with 512×512 pixels per frame. The second video is a real video under in-the-wild conditions for which a high-resolution 3D scan of the captured subject is available: see Fig. 6. In more detail, the real video is a clip from a public talk of the subject and we associate it with a 3D face scan of the same subject under neutral expression that was captured with $DI4D^{TM}$. This video includes 387 frames with 1280×720 pixels per frame. The 3D face scan was acquired with less than 2 months time difference from the day of the public talk, which allows us to reliably consider it as ground truth of the identity component of the 3D facial shape for the real video. In this way, we can quantitatively evaluate how well the 3D facial identity is estimated when different methods are run over the whole video.

6 Experiments

In this section we present in-depth qualitative and quantitative evaluations of our proposed image and video fitting methods. Apart from comparisons with classic and state-of-the-art methods, we are presenting self-evaluations of our fitting framework by comparing results obtained under different settings. We label our "In-The-Wild" image and video fitting methods as *ITW* and *ITW-V* respectively. Further details, visualisations and additional experiments are presented in the Supplementary Material.

We use two different variants of our adopted 3DMM model of shape and texture variation, obtained by using either the Basel Face Model (BFM) [17] or the LSFM model [18] as 3D shape models for identity variation. This is denoted by the labels "(Basel)" and "(LSFM)" after the names of our methods, for example "*ITW(Basel)*" or "*ITW-V(LSFM)*". Note that while LSFM is a more accurate and powerful model, we are also adopting BFM in the experiments for the sake of fairness towards the methods that we compare with, which use BFM or other models of much smaller scale than LSFM. We expand the adopted models for identity variation by incorporating a model for expression variation provided by [16], following the process described in Section 3.1. Regarding the texture component of our 3DMM models that is used by our image fitting method, we trained our "in-the-wild" texture model on the images of iBUG, LFPW & AFW datasets [15] as described in Sec. 3.3 using the 3D shape fits provided by [63].

In the subsequent experimental evaluation, we are comparing with several existing methods for 3DMM fitting as follows:

– "*Classic*": this is an implementation of the classic 3DMM fitting [43], [44] with the original Basel laboratory texture model and full lighting equation.

– "*Linear*": this is the texture-less linear model fitting proposed in [20], [64]. For this method we use the Surrey Model with related blendshapes along with the implementation given in [64].

– "*3DMMedges*": this is the 3DMM fitting method that was recently proposed by Bas et al. [65]. This method is fully automatic and uses landmarks and edge features. For this method, we used the publicly available source code [3] with its default parameters.

– "*Jackson et al. 2017*": this is a very recent method proposed Jackson et al. [66]. It performs 3D face reconstruction from a single image based on Convolutional Neural Networks [66]. It has been reported to achieve promising performance in unconstrained scenarios. To obtain results from this method, we have used the online demo provided by the authors [4].

– "*MoFA*": this is another very recent method, which was proposed by Tewari et al. [67]. It adopts a model-based deep convolutional autoencoder to perform 3D face reconstruction from a single in-the-wild image. Results of this method for a set of in-the-wild images were provided to us by the authors of this method.

– "*4DFace*": in contrast to all previous methods of this list, this method is performing 3DMM fitting on videos rather than images. It was recently introduced by Huber et al. [20], [64]. This the only method for 3DMM fitting on videos with code that is publicly available [5]. We have used the demo app of this code, without making any change on its parameters.

6.1 3DMM fitting on single images

We present both qualitative and quantitative results and comparisons of our proposed "in-the-wild" model on single images.

3. https://github.com/waps101/3DMM_edges
4. http://cvl-demos.cs.nott.ac.uk/vrn/
5. http://www.4dface.org/

Method	*AUC*	*Failure Rate (%)*
ITW	**0.678**	**1.79**
Linear	0.615	4.02
Classic	0.531	13.9

TABLE 1
Accuracy results for facial shape estimation on the KF-ITW database. The table reports the Area Under the Curve (AUC) and Failure Rate of the CEDs of Fig. 8.

Figure 1 demonstrates qualitative results of our image fitting method on a wide range of fits of "in-the-wild" images drawn from the Helen and 300W datasets [14], [15] that qualitatively highlight the effectiveness of the proposed technique. To obtain these results, the BFM model has been used as the identity component of the shape model. We note that in a wide variety of expression, identity, lighting and occlusion conditions our model is able to robustly reconstruct a realistic 3D facial shape that stands up to scrutiny.

Figure 7 shows qualitative comparisons of our ITW method (using LSFM shape identity model) with four existing techniques (MoFA, Jackson et al. 2017, 3DMMedges and Classic) on challenging images of faces under strong expressions. We observe that the results of our method are by far the most visually appealing ones. In contrast to all other tested methods, our method yields 3D face reconstructions that recover both the anatomical characteristics and the facial expressions of the captured subjects in an extremely plausible way, yielding results of unprecedented quality for such challenging conditions.

We also perform a quantitative evaluation on the KF-ITW benchmark, comparing our *ITW(Basel)* method with *Linear* and *Classic* techniques. Fig. 8 shows the Cumulative Error Distribution (CED) for this experiment for the three methods under comparison. Table 1 reports the corresponding Area Under the Curve (AUC) and failure rates. The *Classic* model struggles to fit to the "in-the-wild" conditions present in the test set, and performs the worst. The texture-free *Linear* model does better, but our *ITW(Basel)* model is most able to recover the facial shapes possibly due to its ideal feature basis for the "in-the-wild" conditions.

As a second quantitative evaluation, we employ images of 100 subjects from the Photoface database [68]. We use our *ITW(Basel)* method to find per-pixel normals and compare against two well established Shape-from-Shading (SfS) techniques: *PS-NL* [69] and *IMM* [42]. As a set of four illumination conditions are provided for each subject then we can generate ground-truth facial surface normals using calibrated 4-source Photometric Stereo [70]. In Fig. 9 we show the CED in terms of the mean angular error. *ITW* slightly outperforms *IMM* even though both *IMM* and *PS-NL* use all four available images of each subject.

Apart from in-the-wild conditions like in the previous experiments, we evaluate and compare our fitting method under ideal, laboratory conditions. For this, we use the real images of 3dMD-Lab dataset and compare our *ITW(Basel)* method with *Linear*, *Classic*, *3DMMedges* and *Jackson et al. 2017*. Fig. 10 shows the CED for this experiment. We observe that our method yields a significantly better performance than the compared methods. This suggests that even under more controlled conditions, our image fitting approach is still advantageous over previous approaches.

6.2 3DMM fitting on videos

In addition to the 3D shape recovery of single images we are also evaluating the available techniques on the task of 3D face reconstruction in the videos of 4DMaja dataset as well as in in-the-wild videos collected from the 300VW [71] dataset.

We use a state-of-the-art facial tracker from [72] to fit the videos using a set of sparse landmarks which we use for initialising all the methods. Also, we use the Basel Face Model [17] (BFM) as a standard here to allow a fair comparison across techniques.

In our first experiment, we run ITW-V on 4DMaja(synthetic video) (which provides a ground truth mesh for each frame of the sequence), and compare against "*3DMMedges*" [65], "*4DFace*" [20], [64], "*Classic*" [43], [44] and "*Linear*" [20], [64]. For each examined technique, we calculated an error at each frame of the sequence by computing the average per-vertex error between the recovered mesh and the corresponding ground truth. Fig. 12(a) shows that ITW-V outperforms "*3DMMedges*", which is the second best algorithm, by a large margin. Fig. 12(b) further shows how the per-frame error changes over time. Here, the significantly lower temporal error variance of ITW-V vindicates our decision to regularise identity and enforce smooth expressions over video sequences.

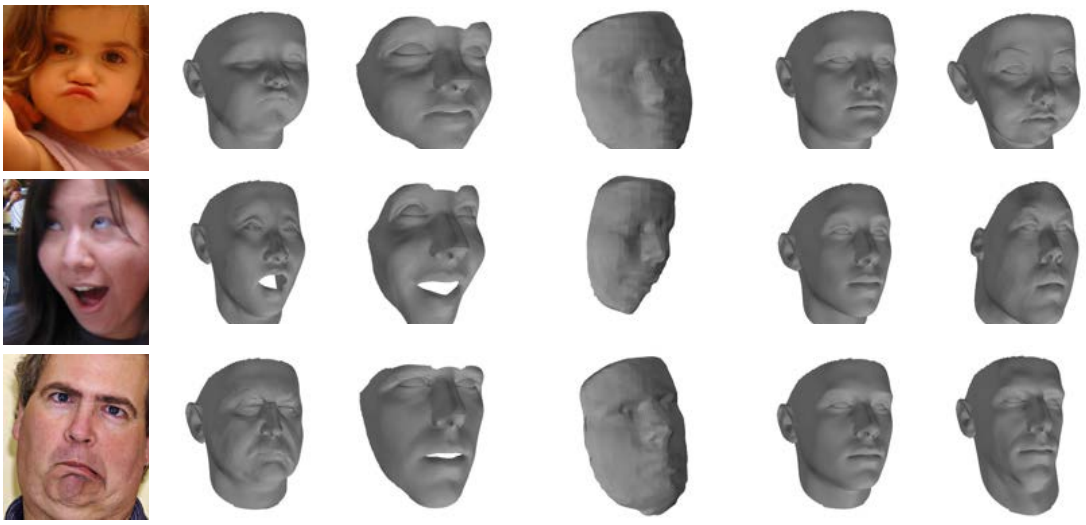
In the next evaluation scenario we run ITW-V on the "in-the-wild" 4DMaja(real video) sequence (which, as a reminder, provides a single ground truth neutral expression mesh). In this case the error is based on comparing the mean recovered mesh for each method across the whole sequence with the single ground truth. In Fig. 13 it can be seen that ITW-V recovers identity more effectively than any other method.

The capability of ITW-V to reconstruct the 3D facial shape in in-the-wild videos is further examined by applying it to videos of the 300VW [71] dataset. For comparison, we both fit our ITW model to each frame individually with no video-specific cost (ITW per frame) and show our full ITW video cost pipeline (ITW-V).Figure 11 shows the representative frames from fitting the videos. We observe that in general both our ITW techniques visually outperform the SfM, Classic and Linear techniques in these challenging videos. We note that ITW-V, our video-specific fitting technique, combines the stability of Structure from Motion (SfM) with the detail from the ITW per frame fitting. The Classic technique's explicit lighting model struggles to model "in-the-wild" effects such as the microphone occlusion (first frame, first video) leading to the algorithm diverging. We note further that ITW-V does not suffer from drift in the identity of the individual (as ITW per frame does, first video) or non-smooth expression changes (see ITW per frame, second video in supplementary material). Finally, we also show in the bottom of this figure how our technique behaves in "in-the-wild" videos when used with the LSFM shape model. We have found this combination of ITW-V with LSFM to be particularly effective, with LSFM providing excellent robustness to variations in age, gender, and ethnicity

A video showing 3D reconstructions from the different methods can be found at https://goo.gl/IcZZWa.

6.3 Self-Evaluation of the proposed method

To decouple the effect on performance of the texture model and the optimisation strategy employed, we present a self-evaluation of our fitting method, where we compare the following:
(i) a full version of our image fitting method (ITW), using the shape variation from BFM [17],
(ii) a version of our image fitting method where we have replaced



(a) Input (b) ITW(LSFM) (c) MoFA (d) Jackson et al. 2017 (e) 3DMMedges (f) Classic

Fig. 7. 3D face reconstruction of challenging face images: qualitative comparison of our method (ITW) with other methods.

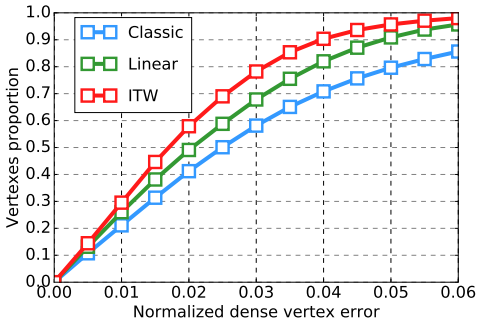


Fig. 8. Accuracy results for facial shape estimation on the KF-ITW database. The results are presented as CEDs of the normalized dense vertex error. Table 1 reports additional measures.

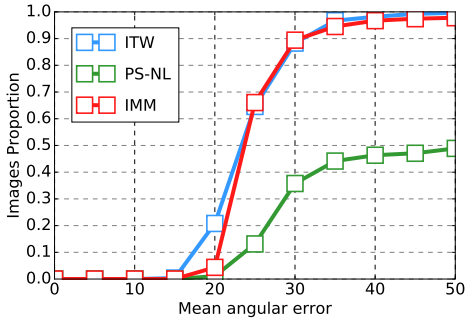


Fig. 9. Accuracy results for facial surface normal estimation on 100 subjects from the Photoface database [68]. The results are presented as CEDs of mean angular error.

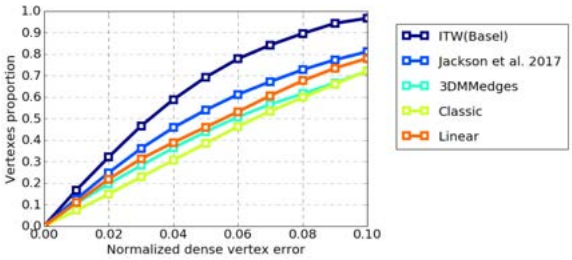


Fig. 10. Facial shape estimation on 3dMD-Lab(real images): quantitative comparison of our image fitting method ITW(Basel) with other methods. The results are presented as CEDs of the normalised dense vertex error.

Method	*AUC*	*Failure Rate (%)*
ITW	**0.632**	**4.30**
RGB-MM	0.610	6.13
Classic	0.545	10.9

TABLE 2
Facial shape estimation on 3dMD-Lab(synthetic images): Quantitative comparison of ITW (our fitting method), RGB-MM (a simplified version of our method where we have replaced the ITW texture model with an RGB texture model) and Classic 3DMM fitting [43]. The table reports the Area Under the Curve (AUC) and Failure Rate of the Cumulative Error Distribution (CED) of each method.

Method	*AUC*	*Failure Rate (%)*
ITW-V	**0.793**	**2.33**
ITW-V, init	0.770	2.46

TABLE 3
3D identity shape estimation on 4DMaja(real video): quantitative self-evaluation of our fitting framework. Comparison of our **video** fitting method (ITW-V) with the initialisation of our video fitting method from sparse landmarks as described in Section 4.2.2 (ITW-V, init). The table reports the Area Under the Curve (AUC) and Failure Rate of the CED of each method.

the learned ITW texture model with the an RGB texture model (laboratory conditions), as provided by BFM [17]. We call this simplified version of our method "RGB-V".

(iii) an implementation of the classic 3DMM fitting ('Classic') [46], which uses the same texture and shape model as in (ii), coming from BFM [17].

This comparison sheds light on the benefits of using an ITW texture model and the proposed energy formulation, independently the one from the other. Table 2 presents the quantitative results of the above three methods on 3dMD-Lab(synthetic images). We observe that method (ii) outperforms method (iii), which suggests that the proposed energy formulation is indeed beneficial as compared to the standard formulation followed by the classic

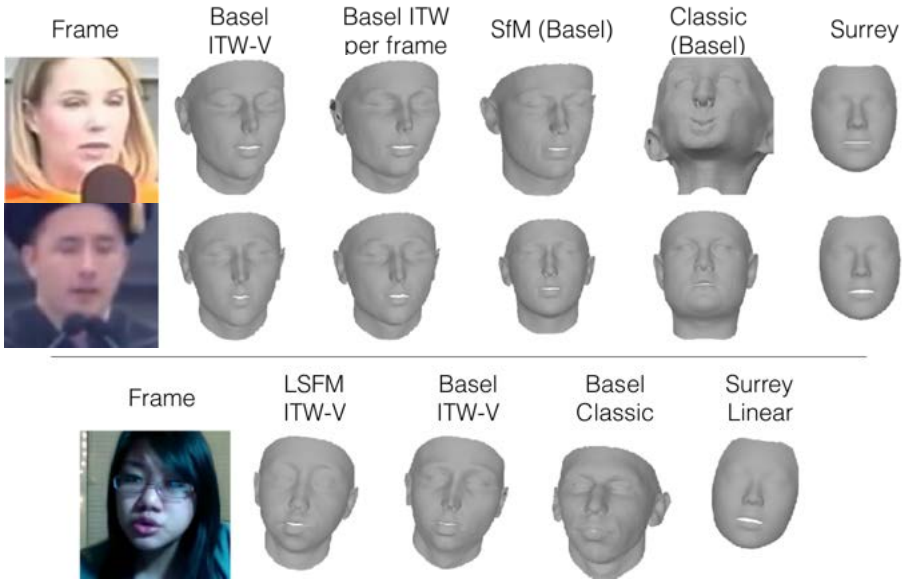
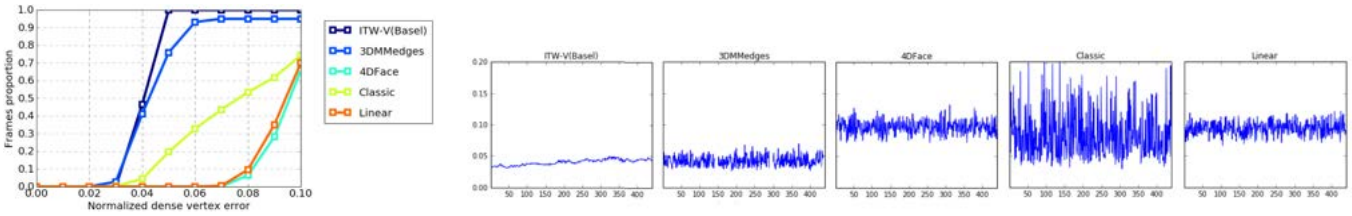


Fig. 11. **Top:** Two sample frames extracted from "in-the-wild" videos along with the 3D reconstructions performed using a variety of techniques. **Bottom:** A final qualitative comparison demonstrating how our proposed technique works well with a range of shape models, including the diverse Large Scale Facial Model (LSFM).



(a) (b)

Fig. 12. 4D facial shape estimation on 4DMaja(synthetic video): quantitative comparison of our video fitting method (ITW-V) with other methods. The results are presented in two ways: a) CEDs of the per-frame mean (over all vertices) normalized dense vertex error, b) Plots of the mean normalized vertex error as a function of time (frame index), where all plots share the same vertical axis.

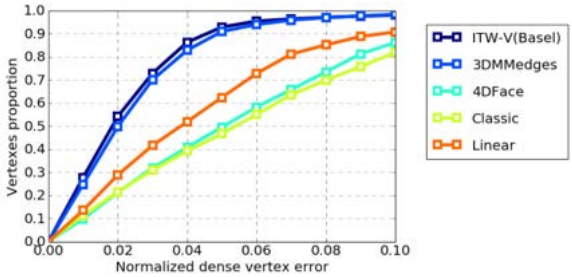


Fig. 13. 3D identity shape estimation on 4DMaja(real video): quantitative comparison of our video fitting method (ITW-V) with other methods. The results are presented as CEDs of the normalized dense vertex error.

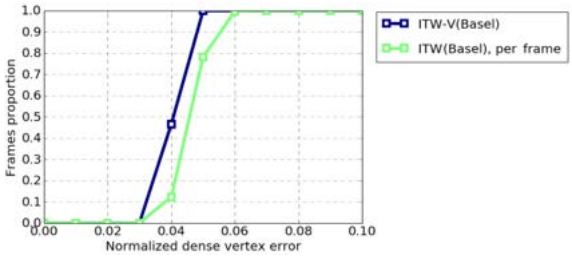


Fig. 14. 4D facial shape estimation on 4DMaja(synthetic video): quantitative self-evaluation of our fitting framework. Comparison of our **video** fitting method (ITW-V(Basel)) with our **image** fitting method applied per-frame (ITW(Basel), per frame), i.e. independently on every frame of the video. The results are presented as CEDs of the per-frame mean (over all vertices) normalized dense vertex error.

3DMM fitting. In addition, we observe that method (i) outperforms method (ii), which suggests that the proposed ITW texture is indeed beneficial as compared to the conventional RGB texture model. A second direction of self-evaluation is to compare our proposed video fitting method ITW-V(Basel) against our image fitting, when the later is applied to the frames of a video independently (ITW(Basel), per frame). We employed this experimental setting to fit 4DMaja(synthetic video) sequence and calculated a mean error at each frame by averaging the differences between the vertexes of the resulting mesh and the ground truth. As presented in Fig. 14 our video fitting outperforms per frame image fitting by a large margin which validates the superiority of our formulation.

Please refer to the Supplementary Material for additional visualisations and self-evaluation experiments.

7 Conclusion

We have presented a novel formulation of 3DMMs re-imagined for use in "in-the-wild" conditions. We capitalise on annotated "in-

the-wild" facial databases to propose a methodology for learning an "in-the-wild" feature-based texture model suitable for 3DMM fitting on images and videos without having to optimise for illumination parameters. We show that we are able to recover shapes with more detail than is possible using purely landmark-driven approaches. Our newly introduced "in-the-wild" datasets, KF-ITW, 4DMaja, & 3dMD-Lab, permit for the first time a quantitative evaluation of 3D facial reconstruction techniques "in-the-wild" on images and videos, and on these evaluations we demonstrate that our in the wild formulation is state-of-the-art, outperforming contemporary 3DMM approaches by a considerable margin.

REFERENCES

[1] V. Jain and E. Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," University of Massachusetts, Amherst, Tech. Rep. UM-CS-2010-009, 2010.

[2] S. Zafeiriou, C. Zhang, and Z. Zhang, "A survey on face detection in the wild: past, present and future," *CVIU*, vol. 138, pp. 1–24, 2015.

[3] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *CVPR*, 2013, pp. 532–539.

[4] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *CVPR*, 2014, pp. 1867–1874.

[5] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, "Incremental face alignment in the wild," in *CVPR*, 2014, pp. 1859–1866.

[6] G. Tzimiropoulos and M. Pantic, "Gauss-newton deformable part models for face alignment in-the-wild," in *CVPR*, 2014, pp. 1851–1858.

[7] S. Zhu, C. Li, C. Change Loy, and X. Tang, "Face alignment by coarse-to-fine shape searching," in *CVPR*, 2015, pp. 4998–5006.

[8] E. Antonakos, J. Alabort-i-Medina, G. Tzimiropoulos, and S. Zafeiriou, "Feature-based lucas-kanade and active appearance models," *TIP*, vol. 24, no. 9, pp. 2617–2632, September 2015.

[9] E. Antonakos, J. Alabort-i-Medina, and S. Zafeiriou, "Active Pictorial Structures," in *CVPR*. Boston, MA, USA: IEEE, June 2015, pp. 5435–5444.

[10] G. Trigeorgis, P. Snape, M. Nicolaou, E. Antonakos, and S. Zafeiriou, "Mnemonic Descent Method: A recurrent process applied for end-to-end face alignment," in *CVPR*. Las Vegas, NV, USA: IEEE, June 2016.

[11] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang, "Interactive facial feature localization," in *ECCV*. Springer, 2012, pp. 679–692.

[12] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *CVPR*. IEEE, 2012, pp. 2879–2886.

[13] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," *IEEE T-PAMI*, vol. 35, no. 12, pp. 2930–2940, 2013.

[14] C. Sagonas, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: The first facial landmark localization challenge," in *ICCV-W*, Sydney, Australia, December 2013.

[15] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image and Vision Computing*, vol. 47, pp. 3–18, 2016.

[16] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," *T-VCG*, vol. 20, no. 3, pp. 413–425, 2014.

[17] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3d face model for pose and illumination invariant face recognition," in *AVSS*. IEEE, 2009, pp. 296–301.

[18] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway, "A 3d morphable model learnt from 10,000 faces," in *CVPR*, 2016.

[19] O. Aldrian and W. A. Smith, "Inverse rendering of faces with a 3d morphable model," *T-PAMI*, vol. 35, no. 5, pp. 1080–1093, 2013.

[20] P. Huber, Z.-H. Feng, W. Christmas, J. Kittler, and M. Rätsch, "Fitting 3D Morphable Face Models using local features," in *ICIP*. IEEE, 2015, pp. 1195–1199.

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1. IEEE, 2005, pp. 886–893.

[22] D. G. Lowe, "Object recognition from local scale-invariant features," in *ICCV*, vol. 2. Ieee, 1999, pp. 1150–1157.

[23] E. Antonakos, J. Alabort-i-Medina, G. Tzimiropoulos, and S. Zafeiriou, "Hog active appearance models," in *ICIP*. IEEE, 2014, pp. 224–228.

[24] G. Papandreou and P. Maragos, "Adaptive and constrained algorithms for inverse compositional active appearance model fitting," in *CVPR*. IEEE, 2008, pp. 1–8.

[25] G. Tzimiropoulos and M. Pantic, "Optimization problems for fast aam fitting in-the-wild," in *ICCV*, 2013, pp. 593–600.

[26] J. Alabort-i Medina and S. Zafeiriou, "A unified framework for compositional fitting of active appearance models," *IJCV*, pp. 1–39, 2016.

[27] J. Alabort-i-Medina, E. Antonakos, J. Booth, P. Snape, and S. Zafeiriou, "Menpo: A comprehensive platform for parametric image alignment and visual deformable models," in *ACM ICM*, ser. MM '14. New York, NY, USA: ACM, 2014, pp. 679–682.

[28] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces." *JOPT*, vol. 4, no. 3, pp. 519–524, Mar. 1987.

[29] A. S. Georghiades, P. N. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *TPAMI*, vol. 23, no. 6, pp. 643–660, 2001.

[30] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *TPAMI*, vol. 25, pp. 218–233, Feb. 2003.

[31] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of cognitive neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[32] R. Ramamoorthi, "Analytic pca construction for theoretical analysis of lighting variability in images of a lambertian object," *TPAMI*, vol. 24, no. 10, pp. 1322–1333, 2002.

[33] R. Ramamoorthi and P. Hanrahan, "On the relationship between radiance and irradiance: determining the illumination from images of a convex lambertian object," *JOPT*, vol. 18, no. 10, pp. 2448–2459, 2001.

[34] A. Shashua, "On photometric issues in 3d visual recognition from a single 2d image," *IJCV*, vol. 21, no. 1-2, pp. 99–122, 1997.

[35] S. R. Marschner, S. H. Westin, E. P. Lafortune, K. E. Torrance, and D. P. Greenberg, "Image-based brdf measurement including human skin," in *Eurographics Workshop on Rendering*. Springer, 1999, pp. 131–144.

[36] P. Snape and S. Zafeiriou, "Kernel-pca analysis of surface normals for shape-from-shading," in *CVPR*, 2014, pp. 1059–1066.

[37] P. L. Worthington and E. R. Hancock, "New constraints on data-closeness and needle map consistency for shape-from-shading," *T-PAMI*, vol. 21, no. 12, pp. 1250–1267, 1999.

[38] W. A. Smith and E. R. Hancock, "Facial shape-from-shading and recognition using principal geodesic analysis and robust statistics," *IJCV*, vol. 76, no. 1, pp. 71–91, 2008.

[39] J. T. Barron and J. Malik, "Shape, illumination, and reflectance from shading," *T-PAMI*, vol. 37, no. 8, pp. 1670–1687, 2015.

[40] W. A. Smith and E. R. Hancock, "Recovering facial shape using a statistical model of surface normal direction," *T-PAMI*, vol. 28, no. 12, pp. 1914–1930, 2006.

[41] P. Snape, Y. Panagakis, and S. Zafeiriou, "Automatic construction of robust spherical harmonic subspaces," in *CVPR*, 2015, pp. 91–100.

[42] I. Kemelmacher-Shlizerman, "Internet based morphable model," in *ICCV*, 2013, pp. 3256–3263.

[43] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3d faces," in *CGIT*. ACM Press/Addison-Wesley Publishing Co., 1999, pp. 187–194.

[44] ——, "Face recognition based on fitting a 3d morphable model," *T-PAMI*, vol. 25, no. 9, pp. 1063–1074, 2003.

[45] S. Romdhani and T. Vetter, "Efficient, robust and accurate fitting of a 3d morphable model." in *ICCV*, vol. 3, 2003, pp. 59–66.

[46] B. Amberg, "Editing faces in videos," Ph.D. dissertation, University_of_Basel, 2011.

[47] S. Romdhani and T. Vetter, "Estimating 3d shape and texture using pixel intensity, edges, specular highlights, texture constraints and a prior," in *CVPR*, vol. 2. IEEE, 2005, pp. 986–993.

[48] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *CVPR*, 2016, pp. 2387–2395.

[49] R. Garg, A. Roussos, and L. Agapito, "A variational approach to video registration with subspace constraints," *IJCV*, vol. 104, no. 3, pp. 286–314, 2013.

[50] ——, "Dense variational reconstruction of non-rigid surfaces from monocular video," in *CVPR*, 2013, pp. 1272–1279.

[51] S. Cheng, I. Marras, S. Zafeiriou, and M. Pantic, "Statistical non-rigid icp algorithm and its application to 3d face alignment," *Image and Vision Computing*, 2016.

[52] J. B. Kuipers *et al.*, *Quaternions and rotation sequences*. Princeton university press Princeton, 1999, vol. 66.

[53] M. Wheeler and K. Ikeuchi, "Iterative estimation of rotation and translation using the quaternion: School of computer science," 1995.

[54] F. Shang, Y. Liu, J. Cheng, and H. Cheng, "Robust principal component analysis with missing data," in *CIKM*, ser. CIKM '14. New York, NY, USA: ACM, 2014, pp. 1149–1158.

[55] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[56] A. Jourabloo and X. Liu, "Large-pose face alignment via cnn-based dense 3d model fitting," in *CVPR*, 2016.
[57] S. Baker and I. Matthews, "Lucas-kanade 20 years on: A unifying framework," *IJCV*, vol. 56, no. 3, pp. 221–255, 2004.
[58] I. Matthews and S. Baker, "Active appearance models revisited," *IJCV*, vol. 60, no. 2, pp. 135–164, 2004.
[59] J. Booth, E. Antonakos, S. Ploumpis, G. Trigeorgis, Y. Panagakis, and S. Zafeiriou, "3d face morphable models "in-the-wild"," in *CVPR*, 2017.
[60] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt *et al.*, "Real-time non-rigid reconstruction using an rgb-d camera," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 4, p. 156, 2014.
[61] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison *et al.*, "Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera," in *UIST*. ACM, 2011, pp. 559–568.
[62] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *ISMAR*. IEEE, 2011, pp. 127–136.
[63] X. Zhu, Z. Lei, X. Liu, H. Shi, and S. Z. Li, "Face alignment across large poses: A 3d solution," in *CVPR*, June 2016.
[64] P. Huber, G. Hu, R. Tena, P. Mortazavian, W. P. Koppen, W. Christmas, M. Rätsch, and J. Kittler, "A multiresolution 3d morphable face model and fitting framework," in *VISAPP*, 2016.
[65] A. Bas, W. A. Smith, T. Bolkart, and S. Wuhrer, "Fitting a 3d morphable model to edges: A comparison between hard and soft correspondences," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 377–391.
[66] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos, "Large pose 3d face reconstruction from a single image via direct volumetric cnn regression," *International Conference on Computer Vision*, 2017.
[67] A. Tewari, M. Zollöfer, H. Kim, P. Garrido, F. Bernard, P. Perez, and T. Christian, "MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction," in *The IEEE International Conference on Computer Vision (ICCV)*, 2017.
[68] S. Zafeiriou, M. Hansen, G. Atkinson, V. Argyriou, M. Petrou, M. Smith, and L. Smith, "The photoface database," in *CVPR*, June 2011, pp. 132–139.
[69] R. Basri, D. Jacobs, and I. Kemelmacher, "Photometric stereo with general, unknown lighting," *IJCV*, vol. 72, no. 3, pp. 239–257, 2007.
[70] D. Marr and H. K. Nishihara, "Representation and recognition of the spatial organization of three-dimensional shapes," *Royal Society of London B: Biological Sciences*, vol. 200, no. 1140, pp. 269–294, 1978.
[71] J. Shen, S. Zafeiriou, G. Chrysos, J. Kossaifi, G. Tzimiropoulos, and M. Pantic, "The first facial landmark tracking in-the-wild challenge: Benchmark and results," in *ICCV-W*, December 2015.
[72] G. Chrysos, E. Antonakos, P. Snape, A. Asthana, and S. Zafeiriou, "A comprehensive performance evaluation of deformable face tracking "in-the-wild"," *IJCV*, 2017.

James Booth is a PhD candidate in the Department of Computing, Imperial College London. His thesis covers the construction and application of highly accurate 3D deformable facial models. James is also an honorary member of the Craniofacial Unit at Great Ormond Street Hospital London.

Anastasios Roussos is a Lecturer (equivalent to Assistant Professor) in Computer Science at the University of Exeter, UK. He is also affiliated with the Department of Computing, Imperial College London. Prior to these positions, he has worked as a postdoctoral researcher at University College London (UCL) (2013-2014) and Queen Mary, University of London (2010-2013).

Evangelos Ververas joined the Intelligent Behavior Understanding Group (IBUG) at the Department of Computing, Imperial College London, in October 2016 and is currently working as a PhD Student/Teaching Assistant under the supervision of Dr. Stefanos Zafeiriou. His research focuses on machine learning and computer vision models for 3D reconstruction and analysis of human faces.

Epameinondas Antonakos is a Computer Vision Research Scientist at the Amazon Development Center in Berlin, Germany. He completed his Ph.D. at the Department of Computing, Imperial College London under the supervision of Dr. Stefanos Zafeiriou focusing on 2D Deformable Models. His research interests lie in the fields of Computer Vision and Statistical Machine Learning.

Yannis Panagakis is a Lecturer in Computer Science at Middlesex University London and a Research Fellow at the Department of Computing, Imperial College London. Yannis received various scholarships and awards for his studies and research, including the prestigious Marie-Curie Fellowship in 2013.

Stefanos Zafeiriou is currently Reader in Machine Learning for Computer Vision with the Department of Computing, Imperial College London, London, U.K, and a Distinguishing Research Fellow with University of Oulu under Finish Distinguishing Professor Programme.