"Look Ma, no landmarks!" – Unsupervised, model-based dense face alignment

Tatsuro Koizumi[1,2] and William A. P. Smith[2]

[1] Canon Inc.
[2] University of York, York, UK
{tk856,william.smith}@york.ac.uk

Abstract. In this paper, we show how to train an image-to-image network to predict dense correspondence between a face image and a 3D morphable model using only the model for supervision. We show that both geometric parameters (shape, pose and camera intrinsics) and photometric parameters (texture and lighting) can be inferred directly from the correspondence map using linear least squares and our novel inverse spherical harmonic lighting model. The least squares residuals provide an unsupervised training signal that allows us to avoid artefacts common in the literature such as shrinking and conservative underfitting. Our approach uses a network that is 10× smaller than parameter regression networks, significantly reduces sensitivity to image alignment and allows known camera calibration or multi-image constraints to be incorporated during inference. We achieve results competitive with state-of-the-art but without any auxiliary supervision used by previous methods.

Keywords: 3D morphable model, dense correspondence, face alignment, landmark, unsupervised, self-supervised

1 Introduction

CNN-based face image analysis with a 3D morphable model (3DMM) [7] has recently shown great promise for both 3D face reconstruction from a single image [22, 31, 24] and dense face alignment [3, 35, 38, 39, 8] (i.e. predicting dense correspondence from image pixels to model). These methods are supervised, limiting their application only to labelled images and not providing a general method that can be extended to new object classes.

One line of CNN-based 3D face reconstruction work offers the promise of overcoming this reliance using model-based autoencoders for self-supervision [29, 13, 10, 28, 6]. Here, a 3DMM and a differentiable renderer are used as a model-based decoder such that a trainable encoder (a CNN) can learn to regress semantically meaningful model parameters. In principle, model-based autoencoders can be trained in a self-supervised fashion. In practice, most rely on auxiliary supervision in the form of landmarks [29, 6, 32], paired identity images [10] or ground truth 3D geometry [13]. The Model-based Face Autoencoder (MoFA) of Tewari et al. [29] did demonstrate a completely unsupervised variant but the estimated

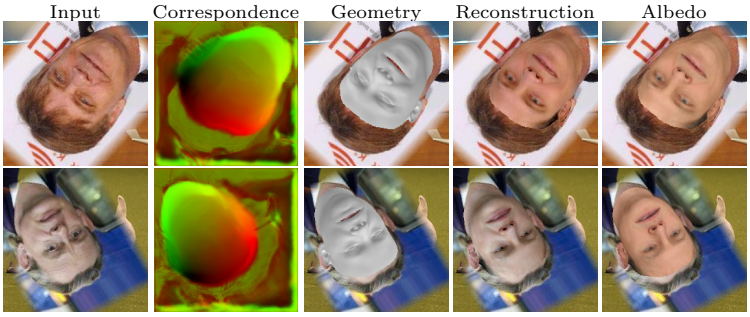


Fig. 1: From a single input image our network learns to predict dense correspondence. From this, we can infer least squares optimal 3DMM geometry and albedo giving high quality reconstructions with 2D transformation invariance.

face is prone to shrinking into the inner face region and requires careful pre-alignment of training images and initialisation of camera parameter predictions such that the initial 3DMM models approximately align with the face images. This makes the approach unable to learn invariance to 2D transformations.

In this paper, we propose a completely unsupervised strategy for learning to fit a 3DMM to a single image. The main difference to previous work is that, instead of image-to-3DMM parameter regression with a contractive CNN, we propose to estimate a dense image-model correspondence map with an image-to-image CNN architecture. There are significant benefits in doing so:

1. All 3DMM parameters can be estimated from a correspondence map (Section 2). Therefore, using a CNN to predict both geometric and photometric parameters, as done in all previous work [29, 13, 10, 28, 6], is redundant.
2. The estimated parameters are least squares optimal with respect to the input image and estimated correspondence map. Optimality for a given image is not guaranteed for a parameter regression CNN whose training objective seeks optimality only in aggregate over the whole training set.
3. Image-to-image CNNs are well suited to estimating correspondence maps with invariance to 2D transformations. Intuitively, it is enough for the correspondence CNN to learn "part detectors" with robustness to 2D rotation (convolution layers are already translation invariant). On the other hand, contractive CNNs are ill-suited to directly regressing geometric parameters with 2D transformation invariance [15]. This is because spatial information is lost in contractive layers and fully connected layers must exhaustively represent both features and their locations to reason about geometric parameters.
4. Image-to-image CNNs are much smaller than parameter regression networks due to the lack of fully connected layers. Concretely, we require $\sim 10\times$ fewer parameters than previous CNN based approaches (e.g. 13.4M parameters for our U-Net versus 138M parameters in VGG-face used by [10, 29]).
5. Every pixel in the input image can contribute to the losses during training. Previous model-based methods learn only from the parts of the image

covered by the geometry of the current 3DMM estimate. In our approach, there is no longer a shortcut for the network to reduce reconstruction loss by shrinking the model to avoid difficult pixels.

6. We defer estimation of actual face geometry. Correspondence is an intermediate representation from which we infer geometry. At test time, if we have access to calibration information or have multiple images from the same camera (e.g. a video), we can exploit these constraints when we finally compute shape from the estimated correspondence map(s). Parameter regression networks cannot do this – they commit to an explanation of the shape and camera parameters for a single image with no way to inject calibration information or constraints post hoc.

Alternatively, our approach can be viewed as a means to learn dense face alignment using model fitting as a form of self-supervision. Correspondence is, in itself, a useful representation. Once trained, the 3DMM can be discarded and the correspondence estimation network used for tasks such as landmarking or semantic segmentation without requiring ground truth labels for supervision.

Our specific novel contributions are as follows. We interpolate a 3DMM to pixel space (Sec. 2.1) then show how to estimate both camera and shape parameters from a correspondence map using linear least squares (Sec. 2.2 and 3.2). We propose an inverse spherical harmonic lighting model enabling simultaneous least squares inverse rendering for both albedo and lighting parameters (Sec. 2.3 and 3.3). Finally, we combine the two least squares solutions with a robust residual loss, a reconstruction loss and priors to enable unsupervised training of our dense alignment network (Sec. 3.4). We make an implementation available[3].

1.1 Related work

Deep integration of 3DMMs The power of deep learning and CNNs has been applied to the task of face model fitting in the last 2 years. Tran et al. [31] use the results of [19] train a CNN discriminatively to regress the same parameters for any single image of the same person. Richardson et al. [22] use synthetic renderings as training data. Both these methods are supervised. MoFA [29] essentially merges analysis-by-synthesis and CNN-based regression in an autoencoder architecture in which the encoder learns the inverse problem, supervised by an appearance error provided by a fixed decoder that implements the forward process. Kim et al. [13] take a similar approach but train on synthetic data that is progressively updated to make it match the distribution of real images. Rather than require that the appearance of the reconstructed model matches that of the input image, Genova et al. [10] use a face encoder to measure similarity in an identity space. Hence, they do not estimate pose or establish correspondence to the input image, but instead ensure discriminative texture and shape are reconstructed. This can be seen as a self-supervised variant of Tran et al. [31]. A number of extension to MoFA have since been considered. Tewari et al. [28]

[3] https://github.com/kzmttr/UMDFA

learn a corrective space to augment the model reconstruction with additional details. Both Tran and Liu [17, 32] and Tewari et al. [27] learn the model itself. In contrast to all of these approaches, we do not regress 3DMM parameters. Instead, we regress an intermediate pixel-wise representation of geometry from which geometric and photometric parameters can be directly inferred in a least squares optimal sense. Importantly, all pixels contribute to this solution, not only those covered by a rendering of the model.

Image-to-image methods Going beyond model fitting, a number of methods make pixel-wise predictions. SFSNet [26] infers lighting and normal and albedo maps from single face images. Their training is bootstrapped using synthetic faces sampled from a model. Sela et al. [25] use an image-to-image network to predict facial depth and correspondence to a canonical model. The network is trained entirely supervised using synthetic data and model fitting requires an offline nonrigid registration to the estimated correspondences. Guler et al. [3] and Yu et al. [35] predict dense correspondence maps using an image-to-image network and supervision provided by landmark-based 3DMM fits. Feng et al. [8] predict a UV map from a 3D face to 2D image coordinates. Zhu et al. [38, 39] propose the projected normalised coordinate code (PNCC) as a representation for dense correspondence. Crispell and Bazik [5] augment PNCC with a predicted 3D offset. All of these approaches are supervised. Several [25, 35, 5] fit a model to estimated depth or correspondence, but this is done as an offline, nonlinear optimisation. In contrast, we show how to fit a 3DMM in-network. This means that we can use the residuals as a supervisory signal for the image-to-image network, negating the need for any direct supervision.

2 3DMM parameters from image-model correspondence

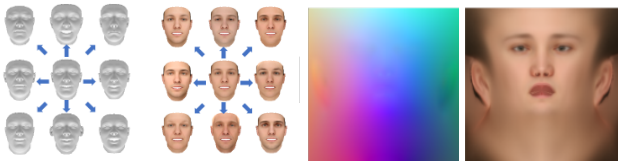
We begin by asking: What can be estimated given dense image-model correspondence alone? Specifically, since we wish to incorporate the estimation process into a network, we are interested in what can be estimated efficiently and in a differentiable manner. Linear least squares satisfies both of these requirements and we use it to estimate optimal geometric and, subsequently, photometric parameters. This necessitates interpolating our 3DMM to pixel space which we explain first.

2.1 Interpolating a 3DMM to UV and pixel space

We represent a 3D face based on a 3DMM:

$$\mathbf{v}_j(\boldsymbol{\alpha}) = \sum_{i=1}^{N_s+N_e} \alpha_i \mathbf{s}_j^i + \bar{\mathbf{s}}_j, \quad \mathbf{r}_j(\boldsymbol{\beta}) = \sum_{i=1}^{N_r} \beta_i \mathbf{a}_j^i + \bar{\mathbf{a}}_j \tag{1}$$

where $\mathbf{v}_j$ is the 3D position and $\mathbf{r}_j$ is the RGB albedo (or reflectance) of the jth vertex respectively. $\mathbf{s}_j^i$ is ith linear basis of the vertex position and $\bar{\mathbf{s}}_j$ is its mean. In the same manner, $\mathbf{a}_j^i$ is ith linear basis of the vertex albedo and



$\mathbf{v}_j$ (a) $\mathbf{r}_j$ (b) $\mathbf{v}_{\boldsymbol{\alpha}}(u,v)$ (c) $\mathbf{r}_{\boldsymbol{\beta}}(u,v)$ (d)

Fig. 2: A 3D morphable model of geometry (a) and albedo (b) can be interpolated to a UV space (c,d) via an embedding. We refer to this as a UV-3DMM.

$\bar{\mathbf{a}}_j$ is its mean. α_i and β_i are the ith coefficient of the linear combination with $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_{N_s+N_e}]^T$ the stacked shape parameters and $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_{N_r}]^T$ the stacked albedo parameters. N_s, N_e and N_r are the number of dimensions for neutral shape, expression and albedo respectively.

UV interpolation of the 3DMM We compute a UV embedding for our 3DMM (in practice by flattening the mean shape – see supplementary material for details) such that every vertex is assigned a fixed 2D UV coordinate. Via barycentric interpolation we can compute a linear shape and texture model for any position, $(u, v) \in [-1, 1]\times[-1, 1]$, in UV space. Accordingly, we write $\mathbf{s}^i(u, v)$, $\bar{\mathbf{s}}(u, v)$, $\mathbf{a}^i(u, v)$ and $\bar{\mathbf{a}}(u, v)$ for the interpolated ith shape basis, shape mean, ith albedo basis and albedo mean at arbitrary location in UV space (u, v). Note that (u, v) is continuous and the barycentric interpolation amounts to taking linear combinations of basis and mean values at the original vertex positions.

The 3D position of the model interpolated at UV coordinate (u, v) is:

$$\mathbf{v}_{\boldsymbol{\alpha}}(u, v) = \mathbf{S}_{u,v}\boldsymbol{\alpha} + \bar{\mathbf{s}}(u, v), \tag{2}$$

where $\mathbf{S}_{u,v} = [\mathbf{s}^1(u, v), \ldots, \mathbf{s}^{N_s+N_e}(u, v)]$ are the stacked shape bases for the model interpolated at UV position (u, v). Similarly, we can write the model albedo interpolated at UV position (u, v):

$$\mathbf{r}_{\boldsymbol{\beta}}(u, v) = \mathbf{A}_{u,v}\boldsymbol{\beta} + \bar{\mathbf{a}}(u, v), \tag{3}$$

where again $\mathbf{A}_{u,v} = [\mathbf{a}^1(u, v), \ldots, \mathbf{a}^{N_r}(u, v)]$ are the stacked albedo bases for the model interpolated at UV position (u, v).

We refer to $\mathbf{v}_{\boldsymbol{\alpha}}(u, v)$ and $\mathbf{r}_{\boldsymbol{\beta}}(u, v)$ as a *UV-3DMM* (see Fig. 2).

UV correspondence map Now, suppose that we are given a correspondence map between a face image, $\mathbf{i}(x, y)$, and the UV space of our 3DMM, i.e. we are given two maps: $u(x, y)$ and $v(x, y)$ defined for each pixel $(x, y) \in \{1, \ldots, W\} \times \{1, \ldots, H\}$ in the face image. Each pixel provides a correspondence between image and model. We can now interpolate our 3DMM at each pixel, via the correspondence map, giving a *pixel-3DMM*: $\mathbf{v}_{\boldsymbol{\alpha}}(u(x, y), v(x, y))$ and $\mathbf{r}_{\boldsymbol{\beta}}(u(x, y), v(x, y))$ (see Fig. 3). Details of how the interpolation is efficiently implemented in-network is described in supplementary material.

$\mathbf{i}(x,y)$ (a) $u(x,y)$ (b) $v(x,y)$ (c) $\mathbf{v}_{\boldsymbol{\alpha}}(u(x,y),v(x,y))$ (d) $\mathbf{r}_{\boldsymbol{\beta}}(u(x,y),v(x,y))$ (e)

Fig. 3: Using estimated correspondences (b,c) from an image (a) to the UV space of the 3DMM, we can define a pixel-3DMM of geometry (d) and albedo (e) in pixel space as a function of 3DMM parameters.

2.2 Least squares shape-from-correspondence

Assume that camera calibration information, i.e. the intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times3}$ and the extrinsic rotation $\mathbf{R} \in \mathbb{R}^{3\times3}$ and translation $\mathbf{t} \in \mathbb{R}^3$, were known. Then, the perspective projection of the 3D position at model UV coordinate (u, v) to pixel position (x, y) is given (up to a scaling) by:

$$\lambda \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \mathbf{project}_{\boldsymbol{\alpha}}(u,v) = \mathbf{K} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \begin{bmatrix} \mathbf{v}_{\boldsymbol{\alpha}}(u,v) \\ 1 \end{bmatrix}, \tag{4}$$

where λ is an arbitrary scale. Using the Direct Linear Transform [11] we can write (4) as a linear system by taking the cross product between the left and right hand sides and setting equal to the zero vector.

Then, the shape parameters, $\boldsymbol{\alpha}$, minimising the reprojection error can be found by solving the following linear least squares problem:

$$\min_{\boldsymbol{\alpha}} \sum_{x=1}^{W} \sum_{y=1}^{H} \left\| \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}_{\times} \mathbf{project}_{\boldsymbol{\alpha}}(u(x,y),v(x,y)) \right\|^2, \text{ where } \begin{bmatrix} \mathbf{x} \end{bmatrix}_{\times} = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}. \tag{5}$$

Note that the residuals of the least squares solution indicate how well the model can explain a shape consistent with the correspondence map and therefore provide a measure of the plausibility of the correspondence map. In practice, $\boldsymbol{\alpha}$ can also be statistically regularised.

During unsupervised training, we of course do not have access to camera calibration information. We later show how to rewrite (5) such that both optimal shape and camera parameters can be found algebraically using linear least squares by additionally estimating a depth map.

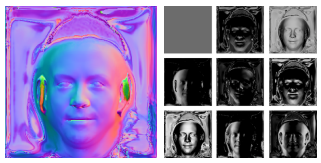
2.3 Least squares inverse rendering

Having computed geometry from correspondence, the surface normals of the shape can be computed. Together with the original image and the correspondence

from image to model, this is sufficient to reason about lighting and albedo. We now show how to simultaneously solve for lighting and albedo coefficients using linear least squares.

Spherical harmonic lighting The spherical harmonic (SH) lighting model [20] efficiently describes how a diffuse object appears under arbitrarily complex environment illumination. At a surface point with normal direction $\mathbf{n}$ and RGB albedo $\mathbf{r}$, the RGB colour intensity, $\mathbf{i}$, is given by:

$$\mathbf{i} = \mathbf{r} \odot \mathbf{B}(\mathbf{n})\mathbf{L}, \tag{6}$$



$\mathbf{n}_{\boldsymbol{\alpha}}(u(x,y),v(x,y))$ $\mathbf{B}\left(\mathbf{n}_{\boldsymbol{\alpha}}(u(x,y),v(x,y))\right)$

(a) (b)

Fig. 4: From shape parameters $\boldsymbol{\alpha}$ we calculate per-vertex surface normals and interpolate via $u(x,y)$ and $v(x,y)$ to a pixel space normal map (a). From this we define an SH basis in pixel space (b).

where $\odot$ denotes element-wise multiplication, $\mathbf{B}(\mathbf{n}) \in \mathbb{R}^{3\times N_L}$ contains the SH basis vectors which depend only on $\mathbf{n}$ and $\mathbf{L} \in \mathbb{R}^{N_L\times 3}$ contains the colour lighting coefficients. For an order 2 approximation, $N_L = 9$ and so there are 27 unknown lighting parameters. This expression is bilinear in diffuse albedo and the spherical harmonic lighting coefficients. This means there is no closed form solution for both optimal albedo and lighting simultaneously. Aldrian and Smith [2] use alternating linear least squares but this requires multiple iterations and is only optimal with respect to the parameters solved for last.

An inverse lighting model In contrast to the conventional model, we use spherical harmonics to represent *inverse lighting*. That is, a quantity that (when multiplied by the image intensity) removes the effect of shading, giving the diffuse albedo. In other words, we use the spherical harmonic basis functions to represent the reciprocal of diffuse shading:

$$\mathbf{i} \odot \mathbf{B}(\mathbf{n})\mathbf{L} = \mathbf{r}. \tag{7}$$

This seemingly subtle difference brings a significant practical advantage: it is linear in both lighting and albedo simultaneously so we can solve for both in a single linear least squares formulation. Importantly, we show empirically in supplementary material that this inverse model can explain conventional SH lighting with very low error.

Inverse rendering with a correspondence map As in the previous section, suppose that we have an estimated correspondence map from a face image to the model. From the geometry estimated by least squares shape-from-correspondence, we can estimate per-vertex surface normals. Then, from the 3DMM UV map we can interpolate a surface normal, $\mathbf{n}_{\boldsymbol{\alpha}}(u,v)$, at any position in UV space or, given the estimated image-model correspondence maps we can interpolate a pixel space normal map $\mathbf{n}_{\boldsymbol{\alpha}}(u(x,y),v(x,y))$ (see Fig. 4(a)). Given

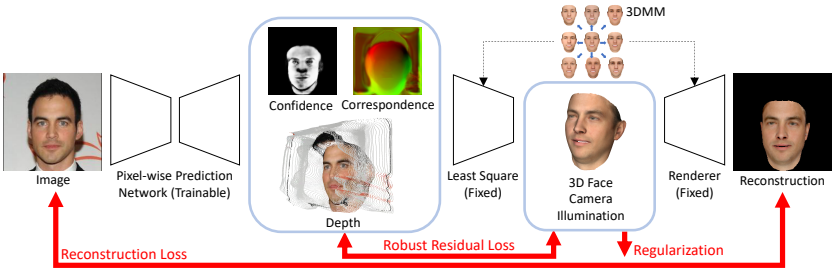


Fig. 5: Overview of proposed architecture. In addition to correspondence our network also predicts a confidence map (for robustness) and depth map (enabling uncalibrated reconstruction). The least squares layer solves first for geometric and then photometric parameters.

the input face image, $\mathbf{i}(x, y)$, we can now write a linear least squares problem for lighting and albedo parameters:

$$\min_{\mathbf{L},\boldsymbol{\beta}} \sum_{x=1}^{W} \sum_{y=1}^{H} \left\| \mathbf{i}(x,y) \odot \mathbf{B}(\mathbf{n}_{\boldsymbol{\alpha}}(u(x,y), v(x,y)))\mathbf{L} - \mathbf{r}_{\boldsymbol{\beta}}(u(x,y), v(x,y)) \right\|^2 . \quad (8)$$

3 Self-supervised learning of dense correspondence

We now show how an image-to-image network for dense face alignment can be trained using self-supervision (see Fig. 5). The idea is that the network predicts a correspondence map from which we implement the fitting process described in Section 2 as differentiable layers. We use a U-Net [23] as the pixel-wise prediction network though any image-to-image architecture would suffice. The network learns from losses measuring the quality of the fit to the correspondence map as well as an appearance loss computed via differentiable rendering. Some modifications are required to incorporate the least squares solutions into the network which we describe in the following sections.

The various loss functions from which the network learns are combined using weights. We distinguish between those that must be manually chosen (i.e. hyperparameters of our method), denoted by η, and those that are learnt as part of the training, denoted by ω.

3.1 Per-pixel confidence

In general, not all of the image will contain face parts. In addition, the face may be occluded by non-face objects such as glasses or unmodelled features such as beards. We do not wish these pixels to contribute to the least squares solutions. Therefore, our network also predicts a scalar confidence map $w(x, y) \in [0, 1]$ indicating whether pixel (x, y) is believed to belong to the face. As with correspondence, this is learnt unsupervised without ever providing the network with ground truth face segmentations.

3.2 Uncalibrated shape-from-correspondence

The least squares solution for geometry in (5) assumed known camera calibration. While this may be available (and can be exploited) at test time, it is not available during unsupervised training. We propose an algebraic solution that allows us to estimate both shape and camera parameters but which requires the network to also estimate a depth map, $z(x, y)$. Again, depth map prediction is learnt unsupervised without any ground truth depth during training. We compute the shape residuals in 3D space by back projection using inverse camera parameters and the estimated depth:

$$\varepsilon_{\text{geo}}(x, y) = \left\| z(x, y)\mathbf{P}[x, y, 1]^T + \mathbf{q} - \mathbf{v}_{\boldsymbol{\alpha}}(u, v) \right\|_2 , \tag{9}$$

where the inverse camera parameters, $\mathbf{P} \in \mathbb{R}^{3\times3}$ and $\mathbf{q} \in \mathbb{R}^3$, are related to standard parameters via $\lambda\mathbf{KR} = \mathbf{P}^{-1}$ and $\lambda\mathbf{Kt} = -\mathbf{P}^{-1}\mathbf{q}$ with λ representing the scale ambiguity. These residuals are linear in the unknown shape parameters and inverse camera parameters.

We can now write the linear least squares system that we solve in-network to compute optimal shape and camera parameters:

$$\boldsymbol{\alpha}^*, \mathbf{P}^*, \mathbf{q}^* = \underset{\boldsymbol{\alpha},\mathbf{P},\mathbf{q}}{\arg\min}\, E_{\text{geo}}(\boldsymbol{\alpha}, \mathbf{P}, \mathbf{q}) + R_{\text{geo}}(\boldsymbol{\alpha}, \mathbf{P}, \mathbf{q}), \tag{10}$$

where $E_{\text{geo}} = \sum_{x,y} w(x, y)\varepsilon_{\text{geo}}(x, y)^2$ is the sum of squared residuals from (9), weighted by the estimated per-pixel confidences and $R_{\text{geo}} = \boldsymbol{\alpha}^T \text{diag}(\boldsymbol{\omega}_{\text{geo}})\boldsymbol{\alpha}$ regularises the solution with the statistical prior, weighting each dimension with a learnable weight.

Since (10) is quadratic, optimal $\boldsymbol{\alpha}$, $\mathbf{P}$, and $\mathbf{q}$ can be obtained using the pseudoinverse matrix. Since the pseudoinverse is differentiable, during training loss gradients can be backpropagated through the least squares solution and into the image-to-image network.

3.3 In-network least squares inverse rendering

With the optimal shape parameters $\boldsymbol{\alpha}^*$ estimated by geometric least squares, we can compute a per-pixel normal map and write the residuals of fitting our inverse lighting model:

$$\varepsilon_{\text{photo}}(x, y) = \left\| \mathbf{i}(x, y) \odot \mathbf{B}(\mathbf{n}_{\boldsymbol{\alpha}*}(u(x, y), v(x, y)))\mathbf{L} - \mathbf{r}_{\boldsymbol{\beta}}(u(x, y), v(x, y)) \right\|_2 . \tag{11}$$

We write a linear least squares system, this time for albedo and lighting:

$$\boldsymbol{\beta}^*, \mathbf{L}^* = \underset{\boldsymbol{\beta},\mathbf{L}}{\arg\min}\, E_{\text{photo}}(\boldsymbol{\beta}, \mathbf{L}) + R_{\text{photo}}(\boldsymbol{\beta}, \mathbf{L}). \tag{12}$$

Once again, $E_{\text{photo}} = \sum_{x,y} w(x, y)\varepsilon_{\text{photo}}(x, y)^2$ is the weighted sum of squared residuals and $R_{\text{photo}} = \boldsymbol{\beta}^T \text{diag}(\boldsymbol{\omega}_{\text{photo}})\boldsymbol{\beta} + \eta_L \|\mathbf{L}\|^2_{\text{Fro}}$ regularises both albedo and lighting parameters. As for geometry, (12) is quadratic and so optimal $\boldsymbol{\beta}$ and $\mathbf{L}$ can be found via the differentiable pseudoinverse.

3.4 Losses

We train our network with four losses (described below):

$$E_{\text{total}} = \eta_{\text{res}} E_{\text{res}} + \eta_{\text{rec}} E_{\text{rec}} + \eta_{\text{stat}} E_{\text{stat}} + \eta_{\text{int}} E_{\text{int}} \tag{13}$$

with $\eta_{\text{rec}} = 1.0$, $\eta_{\text{res}} = 3.0$, $\eta_{\text{stat}} = 1.0$, and $\eta_{\text{int}} = 1.0$.

Least squares residuals loss The least squares layer in our network solves for optimal shape, albedo, camera and lighting parameters by minimising the geometric (9) and photometric (11) residuals. The network can learn from these residuals since they indicate how consistent the 3DMM fit is with the estimated correspondence map (and depth/confidence maps) and the image. Whereas the least squares layer required a closed form solution and therefore uses linear least squares, the loss used for network training is not so constrained. For this reason, we use a robust loss on the residuals:

$$E_{\text{res}} = \sum_{x,y} \min\left(\varepsilon(x,y), 1\right), \quad \text{where } \varepsilon(x,y) = \eta_{\text{geo}} \varepsilon_{\text{geo}}(x,y) + \eta_{\text{photo}} \varepsilon_{\text{photo}}(x,y), \tag{14}$$

and $\eta_{\text{geo}} = 20$ and $\eta_{\text{photo}} = 5$. This loss has an important effect: it encourages the model to expand so that more pixels in the input image can be explained by the model in both geometry and colour. For example, suppose that the pixel-wise network detects an ear with high confidence and estimates good correspondence to the ear region in the model. If the ear of the least squares 3DMM fit is not close to the detected ear pixels, this incurs a residual loss, encouraging the model to expand towards the ear. However, we must make the loss robust since every pixel in the image contributes to it, even background (we do not use the confidence map here). The clamping suppresses the effect from outlier pixels such as occlusion and background.

Reconstruction loss based on differentiable rendering We also compute a conventional reconstruction loss using differentiable rendering to compare the fitted model to the image. Without this, the clamped residual loss does not penalise growing the face to fit to background. We render the 3DMM geometry given by the geometry least squares solution. Our differentiable renderer calculates a projection of each vertex as a 2D point on the image as well as its visibility and RGB albedo. We divide the per-vertex RGB albedo by our inverse lighting model to obtain RGB pixel intensities and measure the discrepancy to the sampled intensities:

$$E_{\text{rec}} = \frac{1}{\sum_{j=1}^{N_v} w_j} \sum_{j=1}^{N_v} w_j \left\| \mathbf{i}(x_j, y_j) - \mathbf{r}_j(\boldsymbol{\beta}^*) \circledcirc \{\mathbf{B}(\mathbf{n}_{\boldsymbol{\alpha}^*}(u(x,y), v(x,y)))\mathbf{L}^*\} \right\|_2, \tag{15}$$

where N_v is the number of the vertices and $w_j = 1$ if a vertex is visible, zero otherwise (computed using self occlusion testing and depth testing against a z-buffer). We use differentiable bilinear sampling and $\mathbf{i}(x_j, y_j)$ represents bilinear

sampling of the input image at the non-integer pixel position (x_j, y_j) given by projection of vertex $\mathbf{v}_j(\boldsymbol{\alpha}^*)$ using the estimated camera parameters.

Statistical regularisation loss This loss encourages the network to keep the estimated face plausible in terms of the shape and albedo parameters. It is the weighted squared average of the estimated 3DMM coefficients:

$$E_{\text{stat}} = \sum_{i=1}^{N_s+N_e} \omega_r^i (\alpha_i^*)^2 + \sum_{i=1}^{N_r} \omega_s^i (\beta_i^*)^2. \tag{16}$$

Since the 3DMM bases are normalised by their standard deviation, the statistical average of α_i^2 and β_i^2 should be kept to be 1 during training. We do this by controlling the loss weight ω_r^i and ω_s^i (see supplementary material).

Camera intrinsics regularisation loss Finally, we employ regularisation on the estimated camera intrinsic parameters. This penalises the difference between vertical and horizontal focal length as well as the shear:

$$E_{\text{int}} = \eta_{\text{asp}} \frac{(k_{11} - k_{22})^2}{k_{11}^2 + k_{22}^2} + \eta_{\text{sh}} \frac{k_{12}^2}{k_{11}^2 + k_{22}^2}, \tag{17}$$

where the k_{ij} are the elements of the intrinsic camera parameter matrix $\mathbf{K}$. The first term represents the difference of vertical and horizontal focal length and the second term represents the sheer component. We normalise the loss by the horizontal and vertical focal length to avoid reducing the scale of focal length. We set $\eta_{\text{asp}} = 1.0$ and $\eta_{\text{sh}} = 1.0$.

4 Training

Initialisation Supervision of our network relies on the difference of appearance between the input image and the estimated face, initial estimation must be enough close to the optimal parameters to obtain meaningful gradient from the loss function. We initialise the network (see supplementary material for details) such that for all inputs it predicts a planar depth map, a correspondence map given by the mean face centred in the image and a binary confidence map given by the rasterisation mask of the centred mean face.

Training data We train on $\sim$ 200k images from pre-aligned CelebA dataset [16]. We augment with random 2D similarity transformations (scale factor: [0.77, 1.3], translation: $[-75, 75]$ pixels horizontal/vertical, rotation $[-180^\circ, 180^\circ]$). The background region is filled by random images from ImageNet[14] with blended boundary. Finally, we crop the image by 224×224 pixels.

Optimisation We use the Adadelta optimizer [36] with learning rate 0.01, batch size 3, 300k iterations. Network weights and biases are initialised by He initialisation [12]. Training takes approximately 120 hours on Nvidia GTX 1080Ti.

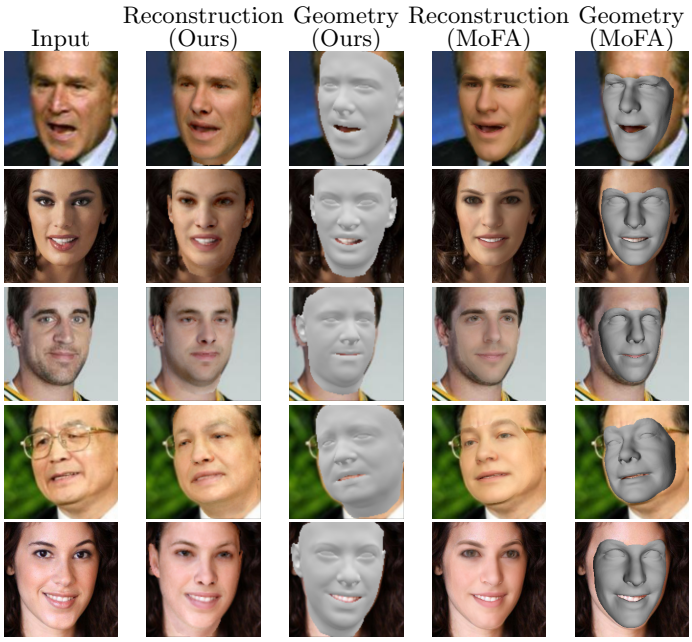


Fig. 6: Result of MoFA [29] and ours from images in MoFA-test dataset.

Fig. 7: Result of multiframe aggregation.

5 Experiments

Qualitative Evaluation We qualitatively evaluate our method based on test images from CelebA dataset (Fig. 6). Our method successfully predicts 3D face including ears under arbitrary 2D similarity transformation. We compare our method with MoFA [29] which can only reconstruct the centre region of a face whereas our method can reconstruct a full head face. Our method also has bet-

	Median	Mean	Std	Supervision
Tran [31]	1.83	2.33	2.05	Fully supervised
PRNet [8]	1.51	1.99	1.90	Fully supervised
RingNet [24]	1.23	1.55	1.32	Landmarks, ID
Ours	1.52	1.89	1.57	None

Table 1: Quantitative evaluation on NoW dataset [24].

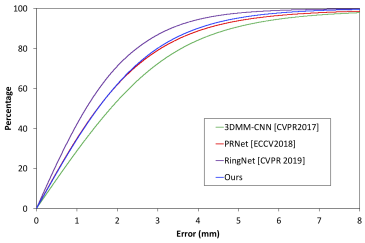


Fig. 8: Cumulative error for the NoW dataset [24].

	Error(HQ)	Error(LQ)	Error(Full)
MTCNN-CNN6-eos [9]	2.70 ± 0.98	2.78 ± 0.95	2.75 ± 0.93
MTCNN-CNN6-3DDFA [9]	2.04 ± 0.67	2.19 ± 0.70	2.14 ± 0.69
SCU-BRL [30]	2.65 ± 0.67	2.87 ± 0.81	2.81 ± 0.80
Ours(w/o E_{int})	2.65 ± 0.98	2.60 ± 0.83	2.62 ± 0.88
Ours	2.39 ± 0.81	2.55 ± 0.82	2.49 ± 0.82

Table 2: Quantitative evaluation on Stirling/ESRC 3D Face Database [1][9].

	AFLW Dataset			AFLW2000-3D Dataset		
Method	Mean[0-30]	Mean[0-90]	Std[0-90]	Mean[0-30]	Mean[0-90]	Std[0-90]
LBF[21]	7.17	17.72	10.64	6.17	16.19	9.87
ESR[4]	5.58	12.07	7.33	4.38	11.72	8.04
CFSS[37]	4.68	12.51	9.49	3.44	13.02	10.08
MDM[33]	5.14	13.40	9.72	4.64	13.07	10.07
SDM[34]	4.67	9.19	6.10	3.56	9.37	7.23
3DDFA[39]	4.11	5.60	0.99	2.84	3.79	1.08
Ours(Direct)	5.51	16.00	10.74	4.98	16.63	10.98
Ours(Fitted)	5.87	18.63	13.20	4.74	18.55	13.38

Table 3: Quantitative evaluation on AFLW[18] and AFLW2000-3D[39] Dataset. The accuracy is evaluated by the Normalized Mean Error.

ter fidelity of reconstruction due to the optimality of the least squares. We also test multiframe aggregation of the pixel-wise prediction (Fig. 7). By optimising multiframe geometry and reflectance to the intermediate output in a single optimisation, superior quality of output can be obtained. See supplementary material for additional qualitative results and comparisons.

Quantitative Evaluation We quantitatively evaluate our method based on landmarks (Tab. 3). We follow the evaluation protocol proposed in Zhu et al. [39] and compare our result with supervised facial landmark detection methods. We evaluate landmarks obtained from both direct correspondence and fitted model. Our network shows comparable result to some supervised methods. We quan-

Fig. 9: Ablation study to show the contribution of intrinsic parameter regularisation E_{int} and robust residual loss E_{res}. We show input, then for each condition we show overlaid reconstruction followed by overlaid geometry.

titatively evaluate our method on the NoW dataset [24] (Tab. 1, Fig. 8) and Stirling/ESRC 3D Face Database (Tab. 2) in which the error of reconstructed neutral face shape is calculated. Our method does not outperform other methods that use richer supervision though it is comparable to some supervised methods.

Ablation Study We investigate the contribution of each loss function qualitatively (Fig. 9) and quantitatively (Tab. 2). The right column in Fig. 9 shows the result trained by only the reconstruction loss and the statistical regularisation. This is a clear example of shrinking problem, and the robust residual loss significantly improves the problem. From Fig. 9 and Tab. 2, it is also clear that the intrinsic parameter regularisation enables the reconstruction of plausible and precise shape.

6 Conclusion

We have presented the first method that combines trainable pixel-wise face alignment with differentiable linear least squares to reconstruct a 3D face model. To the best of our knowledge, this is the first method that enables full ear-to-ear face reconstruction under arbitrary in-plane transformation based on unsupervised training. Our approach has further potential of boosting the performance of conventional supervised face alignment methods by harnessing abundant unlabelled images as well as application to other domains in which annotated images are scarce. In future work, our method can be further improved by incorporating an occlusion model, specular reflection, and perceptual metric to alleviate the vulnerability of photometric error based optimisation. It would also be interesting to make the 3DMM learnable [32] or to estimate a corrective function [28] within our framework allowing reconstruction outside the space of the model.

Acknowledgements

W. Smith is supported by a Royal Academy of Engineering/The Leverhulme Trust Senior Research Fellowship.

References

1. Psychological image collection at stirling (pics), http://pics.stir.ac.uk/
2. Aldrian, O., Smith, W.A.: Inverse rendering of faces with a 3d morphable model. IEEE transactions on pattern analysis and machine intelligence **35**(5), 1080–1093 (2013)
3. Alp Guler, R., Trigeorgis, G., Antonakos, E., Snape, P., Zafeiriou, S., Kokkinos, I.: Densereg: Fully convolutional dense shape regression in-the-wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6799–6808 (2017)
4. Cao, X., Wei, Y., Wen, F., Sun, J.: Face alignment by explicit shape regression. International Journal of Computer Vision **107**(2), 177–190 (2014)
5. Crispell, D., Bazik, M.: Pix2face: Direct 3d face model estimation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2512–2518 (2017)
6. Deng, Y., Yang, J., Xu, S., Chen, D., Jia, Y., Tong, X.: Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In: IEEE Computer Vision and Pattern Recognition Workshops (2019)
7. Egger, B., Smith, W.A., Tewari, A., Wuhrer, S., Zollhoefer, M., Beeler, T., Bernard, F., Bolkart, T., Kortylewski, A., Romdhani, S., et al.: 3d morphable face models–past, present and future. arXiv preprint arXiv:1909.01815 (2019)
8. Feng, Y., Wu, F., Shao, X., Wang, Y., Zhou, X.: Joint 3d face reconstruction and dense alignment with position map regression network. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 534–551 (2018)
9. Feng, Z.H., Huber, P., Kittler, J., Hancock, P., Wu, X.J., Zhao, Q., Koppen, P., Rätsch, M.: Evaluation of dense 3d reconstruction from 2d face images in the wild. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018). pp. 780–786. IEEE (2018)
10. Genova, K., Cole, F., Maschinot, A., Sarna, A., Vlasic, D., Freeman, W.T.: Unsupervised training for 3d morphable model regression. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8377–8386 (2018)
11. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)
12. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015)
13. Kim, H., Zollöfer, M., Tewari, A., Thies, J., Richardt, C., Christian, T.: InverseFaceNet: Deep Single-Shot Inverse Face Rendering From A Single Image. In: Proceedings of Computer Vision and Pattern Recognition (CVPR 2018) (2018)
14. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 1097–1105. Curran Associates, Inc. (2012), http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf
15. Liu, R., Lehman, J., Molino, P., Such, F.P., Frank, E., Sergeev, A., Yosinski, J.: An intriguing failing of convolutional neural networks and the CoordConv solution. In: Advances in Neural Information Processing Systems. pp. 9605–9616 (2018)
16. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (2015)
17. Luan Tran, X.L.: Nonlinear 3d face morphable model. In: IEEE Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT (2018)

18. Martin Koestinger, Paul Wohlhart, P.M.R., Bischof, H.: Annotated Facial Landmarks in the Wild: A Large-scale, Real-world Database for Facial Landmark Localization. In: Proc. First IEEE International Workshop on Benchmarking Facial Image Analysis Technologies (2011)
19. Piotraschke, M., Blanz, V.: Automated 3d face reconstruction from multiple images using quality measures. In: Proc. CVPR. pp. 3418–3427 (2016)
20. Ramamoorthi, R., Hanrahan, P.: An efficient representation for irradiance environment maps. In: Proc. SIGGRAPH. pp. 497–500 (2001)
21. Ren, S., Cao, X., Wei, Y., Sun, J.: Face alignment at 3000 fps via regressing local binary features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1685–1692 (2014)
22. Richardson, E., Sela, M., Kimmel, R.: 3D face reconstruction by learning from synthetic data. In: Proc. 3DV. pp. 460–469 (2016)
23. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. pp. 234–241. Springer (2015)
24. Sanyal, S., Bolkart, T., Feng, H., Black, M.: Learning to regress 3d face shape and expression from an image without 3d supervision. In: Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (Jun 2019)
25. Sela, M., Richardson, E., Kimmel, R.: Unrestricted facial geometry reconstruction using image-to-image translation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1576–1585 (2017)
26. Sengupta, S., Kanazawa, A., Castillo, C.D., Jacobs, D.W.: SfSNet: Learning shape, reflectance and illuminance of faces 'in the wild'. In: Proc. ECCV (2018)
27. Tewari, A., Bernard, F., Garrido, P., Bharaj, G., Elgharib, M., Seidel, H.P., Pérez, P., Zollhöfer, M., Theobalt, C.: FML: Face model learning from videos. arXiv preprint arXiv:1812.07603 (2018)
28. Tewari, A., Zollhöfer, M., Garrido, P., Bernard, F., Kim, H., Pérez, P., Theobalt, C.: Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
29. Tewari, A., Zollöfer, M., Kim, H., Garrido, P., Bernard, F., Perez, P., Christian, T.: MoFA: Model-based Deep Convolutional Face Autoencoder for Unsupervised Monocular Reconstruction. In: The IEEE International Conference on Computer Vision (ICCV) (2017)
30. Tian, W., Liu, F., Zhao, Q.: Landmark-based 3d face reconstruction from an arbitrary number of unconstrained images. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018). pp. 774–779. IEEE (2018)
31. Tran, A.T., Hassner, T., Masi, I., Medioni, G.: Regressing robust and discriminative 3D morphable models with a very deep neural network. In: Proc. CVPR. pp. 5163–5172 (2017)
32. Tran, L., Liu, X.: On learning 3d face morphable model from in-the-wild images. IEEE transactions on pattern analysis and machine intelligence (2019)
33. Trigeorgis, G., Snape, P., Nicolaou, M.A., Antonakos, E., Zafeiriou, S.: Mnemonic descent method: A recurrent process applied for end-to-end face alignment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4177–4187 (2016)
34. Yan, J., Lei, Z., Yi, D., Li, S.Z.: Learn to combine multiple hypotheses for accurate face alignment. 2013 IEEE International Conference on Computer Vision Workshops pp. 392–396 (2013)

35. Yu, R., Saito, S., Li, H., Ceylan, D., Li, H.: Learning dense facial correspondences in unconstrained images. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4723–4732 (2017)
36. Zeiler, M.D.: Adadelta: An adaptive learning rate method. CoRR **abs/1212.5701** (2012)
37. Zhu, S., Li, C., Change Loy, C., Tang, X.: Face alignment by coarse-to-fine shape searching. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4998–5006 (2015)
38. Zhu, X., Lei, Z., Liu, X., Shi, H., Li, S.Z.: Face alignment across large poses: A 3d solution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 146–155 (2016)
39. Zhu, X., Liu, X., Lei, Z., Li, S.Z.: Face alignment in full pose range: A 3d total solution. IEEE transactions on pattern analysis and machine intelligence **41**(1), 78–92 (2017)