

# Unsupervised Learning of Probably Symmetric Deformable 3D Objects From Images in the Wild (Invited Paper)

Shangzhe Wu<sup>ID</sup>, Christian Rupprecht<sup>ID</sup>, and Andrea Vedaldi<sup>ID</sup>

**Abstract**—We propose a method to learn 3D deformable object categories from raw single-view images, without external supervision. The method is based on an autoencoder that factors each input image into depth, albedo, viewpoint and illumination. In order to disentangle these components without supervision, we use the fact that many object categories have, at least approximately, a symmetric structure. We show that reasoning about illumination allows us to exploit the underlying object symmetry even if the appearance is not symmetric due to shading. Furthermore, we model objects that are probably, but not certainly, symmetric by predicting a symmetry probability map, learned end-to-end with the other components of the model. Our experiments show that this method can recover very accurately the 3D shape of human faces, cat faces and cars from single-view images, without any supervision or a prior shape model. On benchmarks, we demonstrate superior accuracy compared to another method that uses supervision at the level of 2D image correspondences.

**Index Terms**—Unsupervised 3D reconstruction, single-image 3D reconstruction, intrinsic image decomposition

## 1 INTRODUCTION

THE ability to understand and reconstruct the content of images in 3D is of great importance in many computer vision applications. Yet, when it comes to learning categories of visual objects, for instance to detect and segment them, most approaches model them as 2D patterns [1], with no obvious understanding of their 3D structure. Thus, in this paper we consider the problem of learning categories of 3D deformable objects. Furthermore, we do so under two challenging conditions. The first condition is that *no 2D or 3D ground truth information* (such as keypoints, segmentation, depth maps, or prior knowledge of a 3D model) is available. Learning without external supervisions removes the need for collecting image annotations, which is often a major obstacle to deploying deep learning to new applications. The second condition is that learning can only use an *unconstrained collection of single-view images* — in particular, it does not use multiple views of the same object instance. Learning from single-view images is useful because in many applications we only have a source of independent still images to work with (for example obtained from an Internet search engine).

In more detail, we introduce a new learning algorithm that takes as input a collection of single-view images of a deformable object category and produces as output a deep

network that can estimate the 3D shape of any object instance given a single image of it (Fig. 1). The algorithm is based on an autoencoder that internally decomposes the image into albedo, depth, illumination and viewpoint, *without direct supervision for any of these factors*. In general, decomposing images into these four factors is ill-posed. We thus seek for a minimal set of assumptions that makes the problem solvable. To this end, we note that many object categories are *symmetric* (e.g. almost all animals and many handcrafted objects). If an object is perfectly symmetric, mirroring any image of it results in a second virtual view of the object. Furthermore, if point correspondences between the image and its mirrored version can be established, then the 3D shape of the object can be recovered using any of a number of standard multi-view 3D reconstruction approaches [2], [3], [4], [5], [6]. Motivated by this, we seek to leverage symmetry as a cue to constrain this decomposition task.

While symmetry is a powerful cue, using it in practice is far from trivial. First, even if symmetry allows to obtain a pair of virtual views of an object, reconstruction still require to establish point correspondences between them, which can be difficult to do in an unsupervised manner. For instance, the appearance of symmetric points may still differ substantially due to asymmetric illumination. Second, specific object instances are in practice never fully symmetric, neither in shape nor appearance. Shape is non-symmetric due to variations in pose or other details (e.g. the hair style or expressions in a human face), and albedo can also be non-symmetric (e.g. asymmetries in the texture of cat's fur).

We address these issues in two ways. First, we explicitly account for the effect of illumination in the reconstruction pipeline by decomposing the appearance into albedo and shading. In this manner, the model learns to explain

• The authors are with the Department of Engineering Science, University of Oxford, OX1 2JD Oxford, U.K. E-mail: {szwu, chrisr, vedaldi}@robots.ox.ac.uk.

Manuscript received 20 Dec. 2020; accepted 16 Apr. 2021. Date of publication 29 Apr. 2021; date of current version 6 Mar. 2023.

(Corresponding author: Shangzhe Wu.)

Recommended for acceptance by Silvio Savarese and Ce Liu.

Digital Object Identifier no. 10.1109/TPAMI.2021.3076536

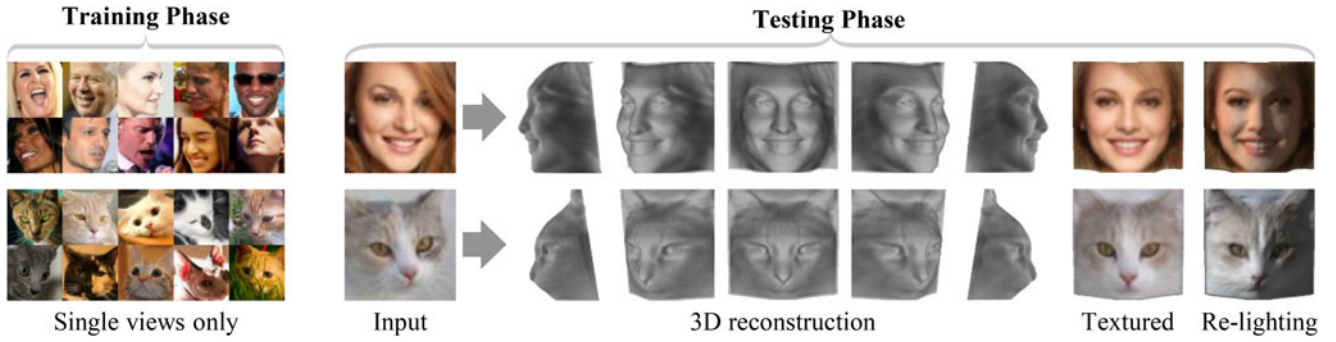


Fig. 1. *Unsupervised learning of 3D deformable objects from in-the-wild images.* Left: Training uses *only* single views of the object category with *no* additional supervision at all (i.e. no ground-truth 3D information, multiple views, or any prior model of the object). Right: Once trained, our model reconstructs the 3D pose, shape, albedo and illumination of a deformable object instance from a single image with excellent fidelity.

asymmetries in the object appearance resulting from illumination, allowing it to better understand how pairs of symmetric views of the object correspond. Moreover, since shading provides information on the surface normals and thus the 3D shape, decomposing it allows the model to explicitly use this information to constrain 3D shapes. Second, we augment the model to reason about potential lack of symmetry in the objects. To do this, the model predicts, along with the factors listed above, a dense map explaining the probability that a given pixel has a symmetric counterpart in the image.

We combine these elements in an end-to-end learning formulation, where all components, including the symmetry probability map, are learned from raw RGB images only. As a further contribution, we show that, rather than enforcing the symmetry by adding further terms to the learning objective, we can instead do so indirectly. The latter is obtained by randomly mirroring the internal representation of the object, thus encouraging the autoencoder to generate a symmetric view of the object. The advantage of this approach is that it avoids the need to introduce and thus tune additional terms in the learning objective.

We test our method on several datasets, including human faces, cat faces and synthetic cars. We provide a thorough ablation study and extensive analyses using a synthetic face dataset with the necessary 3D ground truth. On real images, we achieve higher fidelity reconstruction results compared to other methods [7], [8] that do not rely on 2D or 3D ground truth information, nor prior knowledge of a 3D model of the instance or class. In addition, our method outperforms a recent state-of-the-art method [9] that uses keypoint supervision for 3D reconstruction on real faces, while our method uses no external supervision at all. As a by-product, our method also learns intrinsic image decomposition without any external supervision. Finally, we demonstrate that our trained model generalizes to non-natural images, such as paintings and cartoon drawings, as well as video frames without any fine-tuning.

This article is an extension and archival version of our previous work [10]. In this article, we expand the literature review, provide additional technical details, and include additional experiments and discussions that reveal the important insights of the proposed algorithm, including how it works, how it may fail, and how it compares to prominent model-based methods on 3D reconstruction

benchmarks. The code and pretrained models are available at <https://github.com/elliottwu/unsup3d>.

## 2 RELATED WORK

In order to assess our contribution in relation to the vast literature on image-based 3D reconstruction, it is important to consider three aspects of each approach: which information is used, which assumptions are made, and what the output is. Below and in Table 1 we compare our contribution to prior works based on these factors.

TABLE 1  
Comparison With Selected Prior Work: Supervision, Goals, and Data

Paper	Supervision	Goals	Data
[11]	3D scans	3DMM	Face
[12]	3DV, I	Prior on 3DV	ShapeNet, Ikea
[13]	3DP	Prior on 3DP	ShapeNet
[14]	3DM	Prior on 3DM	Face
[15]	3DMM, 2DKP, I	Refine 3DMM fit to I	Face
[16]	3DMM, 2DKP, I	Fit 3DMM to I+2DKP	Face
[17]	3DMM	Fit 3DMM to 3D scans	Face
[18]	3DMM, 2DKP	Pred. 3DMM	Humans
[19]	3DMM, 2DS+KP	Pred. N, A, L	Face
[20]	3DMM, I	Pred. 3DM, VP, T, E	Face
[21]	3DMM, 2DKP, I	Fit 3DMM to I	Face
[22]	2DS	Prior on 3DV	ModelNet
[23]	2DS	Pred. 3DV	ShapeNet
[24]	I, 2DS, VP	Prior on 3DV	ShapeNet, PAS3D
[25]	I, 2DS+KP	Pred. 3DM, T, VP	Birds
[26]	I, 2DS	Pred. 3DM, T, L, VP	ShapeNet, Birds
[27]	I, 2DS	Pred. 3DV, VP	ShapeNet, others
[28] <sup>*</sup>	I, 2DS, 3DTM	Fit 3DTM to I	Animals
[29] <sup>*</sup>	I, 2DS, 3DTM	Pred. 3DM, T, VP	Birds, PAS3D
[30] <sup>*</sup>	I, 2DS <sup>†</sup>	Pred. 3DM, T, VP	Birds, PAS3D
[8]	I	Prior on 3DM, T	Face
[31]	I	Prior on 3DV, T	Face, others
[32] <sup>*</sup>	I	Prior on 3DV, T	ShapeNet, others
[7]	I	Pred. 3DM, VP, T <sup>‡</sup>	Face
[33]	I	Pred. V, L, VP	ShapeNet
Ours	I	Pred. D, L, A, VP	Face, others

I: image, 3DMM: 3D morphable model, 3DTM: 3D template model, 2DKP: 2D keypoints, 2DS: 2D silhouette, 3DP: 3D points, VP: viewpoint, E: expression, 3DM: 3D mesh, 3DV: 3D volume, D: depth, N: normals, A: albedo, T: texture, L: light, PAS3D: PASCAL 3D+ [34]. <sup>†</sup> in the form of part segmentation maps. <sup>‡</sup> can also recover A and L in post-processing. <sup>\*</sup> appear after our original paper was published.

Our method uses single-view images of an object category as training data, assumes that the objects belong to a specific class (e.g. human faces) which is weakly symmetric, and outputs a monocular predictor capable of decomposing any image of the category into shape, albedo, illumination, viewpoint and symmetry probability.

## 2.1 Structure From Motion

Traditional methods such as Structure from Motion (SfM) [35] can reconstruct the 3D structure of individual rigid scenes given as input multiple views of each scene and 2D keypoint matches between the views. This can be extended in two ways. First, *monocular reconstruction methods* can perform dense 3D reconstruction from a single image without 2D keypoints [36], [37], [38]. However, they require multiple views [38] or videos of rigid scenes for training [36]. Second, *Non-Rigid SfM* (NRSfM) approaches [39], [40] can learn to reconstruct deformable objects by allowing 3D points to deform in a limited manner between views, but require supervision in terms of annotated 2D keypoints for both training and testing. Hence, neither family of SfM approaches can learn to reconstruct deformable objects from raw pixels of a single view.

## 2.2 Shape From X

Many other monocular cues have been used as alternatives or supplements to SfM for recovering shape from images, such as shading [41], [42], silhouettes [43], texture [44], symmetry [2], [3] etc. In particular, our work is inspired from *shape from symmetry* and *shape from shading*. Shape from symmetry [2], [3], [4], [5] reconstructs symmetric objects from a single image by using the mirrored image as a virtual second view, provided that symmetric correspondences are available. [5] also shows that it is possible to detect symmetries and correspondences using descriptors. Shape from shading [41], [42] assumes a shading model such as Lambertian reflectance, and reconstructs the surface by exploiting the non-uniform illumination.

## 2.3 Category-Specific Reconstruction

Learning-based methods have recently been leveraged to reconstruct objects from a single view, either in the form of a raw image or 2D keypoints (see also Table 1). While this task is ill-posed, it has been shown to be solvable by learning a suitable object prior from the training data [11], [12], [13], [14]. A variety of supervisory signals have been proposed to learn such priors. Besides using 3D ground truth directly, authors have considered using videos [36], [45], [46], [47], [48], stereo pairs [38], [49] and multi-view images [50], [51], [52], [53], [54].

Other approaches have used single views with 2D keypoint annotations [9], [25], [55], [56] or object masks [23], [25], [26], [29]. For objects such as human bodies and human faces, some methods [16], [17], [18], [20], [21], [29], [57], [58], [59], [60], [61] have learned to reconstruct from raw images, but starting from the knowledge of a predefined shape model, such as SMPL [62] or Basel [11], or shape templates. These prior models are constructed using specialized hardware and/or other forms of supervision, which are often difficult to obtain for deformable objects in the wild, such as animals, and also limited in details of the shape.

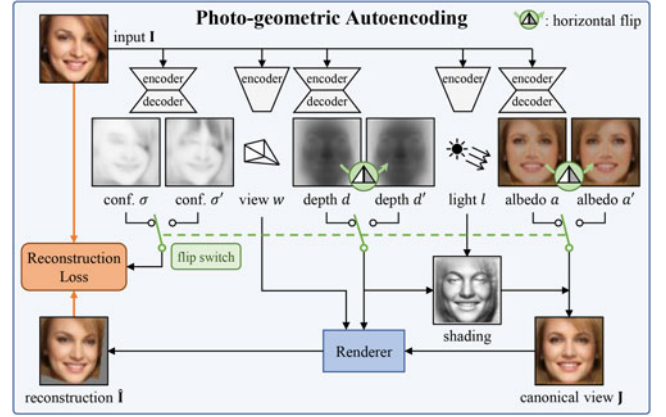


Fig. 2. *Photo-geometric autoencoding*. Our network  $\Phi$  decomposes an input image  $I$  into depth, albedo, viewpoint and lighting, together with a pair of confidence maps. It is trained to reconstruct the input without external supervision.

Only recently have authors attempted to learn the geometry of object categories from raw, monocular views *only*. Thewlis *et al.* [63], [64] uses equivariance to learn dense landmarks, which recovers the 2D geometry of the objects. DAE [65] learns to predict a deformation field through heavily constraining an autoencoder with a small bottleneck embedding and lift that to 3D in [7] — in post processing, they further decompose the reconstruction in albedo and shading, obtaining an output similar to ours.

Adversarial learning has been proposed as a way of hallucinating new views of an object. Some of these methods start from 3D representations [12], [13], [14], [32], [66]. Kato *et al.* [24] trains a discriminator on raw images but uses viewpoint as additional supervision. HoloGAN [31] only uses raw images but does not obtain an explicit 3D reconstruction. Szabo *et al.* [8] uses adversarial training to reconstruct 3D meshes of the object, but does not assess their results quantitatively. Henzler *et al.* [27] also learns from raw images, but only experiments with images that contain the object on a white background, which is akin to supervision with 2D silhouettes. In Section 4.4, we compare to [7], [8] and demonstrate superior reconstruction results with much higher fidelity.

Since our model generates images from an internal 3D representation, one essential component is a differentiable renderer. However, with a traditional rendering pipeline, gradients across occlusions and boundaries are not defined. Several soft relaxations have thus been proposed [67], [68], [69]. Here, we use a PyTorch implementation<sup>1</sup> of [68].

## 3 METHOD

Our learning algorithm, illustrated in Fig. 2, takes as input a collection of independent images of objects of a certain category, such as human or cat faces. It then produces as output a model  $\Phi$  that, given any new image, recovers the object's 3D shape, albedo, illumination and viewpoint.

As the algorithm has only raw images to learn from, the learning objective is reconstructive: namely, the model is trained so that the combination of the four factors gives

1. [https://github.com/daniilidis-group/neural\\_renderer](https://github.com/daniilidis-group/neural_renderer)

back the input image. This results in an auto-encoding pipeline where the factors have, due to the way they are combined to generate an image, an specific photo-geometric interpretation.

Due to the lack of 2D or 3D supervision and of a 3D prior on the possible shapes of the objects, this reconstruction problem is ill-posed. In order to address this issue, we use the fact that many object categories are *bilaterally symmetric*, which provides a strong geometric cue to remove the most severe reconstruction ambiguities. In practice, the appearance of specific object instances is never exactly symmetric due to deformations of the 3D shape and asymmetric details in the shape itself as well as in the illumination and albedo. We take two measures to account for these asymmetries. First, we explicitly model asymmetric illumination. Second, our model also estimates, for each pixel in the input image, a confidence score that explains the probability of the pixel having a symmetric counterpart in the image (denoted as *conf*.  $\sigma$  and  $\sigma'$  in Fig. 2).

The following sections describe how this is done, looking first at the photo-geometric autoencoder (Section 3.1), then at how symmetries are modelled (Section 3.2), followed by details of the image formation (Section 3.3) and the optional use of a perceptual loss (Section 3.4).

### 3.1 Photo-Geometric Autoencoding

An image  $\mathbf{I}$  is a function  $\Omega \rightarrow \mathbb{R}^3$  defined on a grid  $\Omega = \{0, \dots, W-1\} \times \{0, \dots, H-1\}$ , or, equivalently, a tensor in  $\mathbb{R}^{3 \times W \times H}$ . We assume that the image is roughly centered on an instance of the object of interest. The goal is to learn a function  $\Phi$ , implemented as a neural network, that maps the image  $\mathbf{I}$  to four factors  $(d, a, w, l)$  comprising a *depth map*  $d: \Omega \rightarrow \mathbb{R}_+$ , an *albedo image*  $a: \Omega \rightarrow \mathbb{R}^3$ , a *global light direction*  $l \in \mathbb{S}^2$ , and a *viewpoint*  $w \in \mathbb{R}^6$  so that the image can be reconstructed from them.

The image  $\mathbf{I}$  is reconstructed from the four factors in two steps, *lighting*  $\Lambda$  and *reprojection*  $\Pi$ , as follows:

$$\hat{\mathbf{I}} = \Pi(\Lambda(a, d, l), d, w). \quad (1)$$

The lighting function  $\Lambda$  generates a version of the object based on the depth map  $d$ , the light direction  $l$  and the albedo  $a$  as seen from a canonical viewpoint  $w = 0$ . The viewpoint  $w$  represents the transformation between the canonical view and the viewpoint of the actual input image  $\mathbf{I}$ . Then, the reprojection function  $\Pi$  simulates the effect of a viewpoint change and generates the image  $\hat{\mathbf{I}}$  given the canonical depth  $d$  and the shaded canonical image  $\Lambda(a, d, l)$ . Learning uses a reconstruction loss which encourages  $\mathbf{I} \approx \hat{\mathbf{I}}$  (Section 3.2).

#### 3.1.1 Discussion

The effect of lighting could be incorporated in the albedo  $a$  by interpreting the latter as a texture rather than as the object's albedo. However, there are two good reasons to avoid this. First, the albedo  $a$  is often symmetric even if the illumination causes the corresponding appearance to look asymmetric. Separating them allows us to more effectively incorporate the symmetry constraint described below. Second, shading provides an additional cue on the underlying 3D shape [70], [71]. In particular, unlike the recent work

of [65] where a shading map is predicted independently from shape, our model computes the shading based on the predicted depth, mutually constraining each other.

### 3.2 Probably Symmetric Objects

Leveraging symmetry for 3D reconstruction requires identifying symmetric object points in an image. Here we do so implicitly, assuming that depth and albedo, which are reconstructed in a canonical frame, are symmetric about a fixed vertical plane. An important beneficial side effect of this choice is that it helps the model discover a 'canonical view' for the object, which is important for reconstruction [40].

To do this, we consider the operator that flips a map  $a \in \mathbb{R}^{C \times W \times H}$  along the horizontal axis:<sup>2</sup>  $[\text{flip}a]_{c,u,v} = a_{c,W-1-u,v}$ . We then require  $d \approx \text{flip}d'$  and  $a \approx \text{flip}a'$ . While these constraints could be enforced by adding corresponding loss terms to the learning objective, they would be difficult to balance. Instead, we achieve the same effect indirectly, by obtaining a second reconstruction  $\hat{\mathbf{I}}'$  from the flipped depth and albedo

$$\hat{\mathbf{I}}' = \Pi(\Lambda(a', d', l), d', w), \quad a' = \text{flip } a, \quad d' = \text{flip } d. \quad (2)$$

Then, we consider two reconstruction losses encouraging  $\mathbf{I} \approx \hat{\mathbf{I}}$  and  $\mathbf{I} \approx \hat{\mathbf{I}}'$ . Since the two losses are commensurate, they are easy to balance and train jointly. Most importantly, this approach allows us to easily reason about symmetry probabilistically, as explained next.

The source image  $\mathbf{I}$  and the reconstruction  $\hat{\mathbf{I}}$  are compared via the loss

$$\mathcal{L}(\hat{\mathbf{I}}, \mathbf{I}, \sigma) = -\frac{1}{|\Omega|} \sum_{uv \in \Omega} \ln \frac{1}{\sqrt{2}\sigma_{uv}} \exp -\frac{\sqrt{2}\ell_{1,uv}}{\sigma_{uv}}, \quad (3)$$

where  $\ell_{1,uv} = |\hat{\mathbf{I}}_{uv} - \mathbf{I}_{uv}|$  is the  $L_1$  distance between the intensity of pixels at location  $uv$ , and  $\sigma \in \mathbb{R}_+^{W \times H}$  is a *confidence map*, also estimated by the network  $\Phi$  from the image  $\mathbf{I}$ , which expresses the *aleatoric uncertainty* of the model. The loss can be interpreted as the negative log-likelihood of a factorized Laplacian distribution on the reconstruction residuals. Optimizing likelihood causes the model to self-calibrate, learning a meaningful confidence map [72].

Modelling uncertainty is generally useful, but in our case is particularly important when we consider the "symmetric" reconstruction  $\hat{\mathbf{I}}'$ , for which we use the same loss  $\mathcal{L}(\hat{\mathbf{I}}', \mathbf{I}, \sigma')$ . Crucially, we use the network to estimate, also from the same input image  $\mathbf{I}$ , a *second* confidence map  $\sigma'$ . This confidence map allows the model to learn which portions of the input image might *not* be symmetric. For instance, in some cases hair on a human face is not symmetric, as shown in Fig. 2, and  $\sigma'$  can assign a higher reconstruction uncertainty to the hair region where the symmetry assumption is not satisfied. Note that this depends on the *specific* instance under consideration, and is learned by the model itself.

Overall, the learning objective is given by the combination of the two reconstruction errors

$$\mathcal{E}(\Phi; \mathbf{I}) = \mathcal{L}(\hat{\mathbf{I}}, \mathbf{I}, \sigma) + \lambda_f \mathcal{L}(\hat{\mathbf{I}}', \mathbf{I}, \sigma'), \quad (4)$$

2. The choice of axis is arbitrary as long as it is fixed.

where  $\lambda_f = 0.5$  is a weighing factor,  $(d, a, w, l, \sigma, \sigma') = \Phi(\mathbf{I})$  is the output of the neural network, and  $\hat{\mathbf{I}}$  and  $\hat{\mathbf{I}}'$  are obtained according to Eqs. (1) and (2).

### 3.3 Image Formation Model

We now describe the functions  $\Pi$  and  $\Lambda$  in Eq. (1) in more detail. The image is formed by a camera looking at a 3D object. If we denote with  $P = (P_x, P_y, P_z) \in \mathbb{R}^3$  a 3D point expressed in the reference frame of the camera, this is mapped to pixel  $p = (u, v, 1)$  by the following projection:

$$p \propto KP, \quad K = \begin{bmatrix} f & 0 & c_u \\ 0 & f & c_v \\ 0 & 0 & 1 \end{bmatrix}, \quad \begin{cases} c_u = \frac{W-1}{2}, \\ c_v = \frac{H-1}{2}, \\ f = \frac{W-1}{2 \tan \frac{\theta_{\text{FOV}}}{2}}. \end{cases} \quad (5)$$

This model assumes a perspective camera with *field of view* (FOV)  $\theta_{\text{FOV}}$ . We assume a nominal distance of the object from the camera at about 1m. Given that the images are cropped around a particular object, we assume a relatively narrow FOV of  $\theta_{\text{FOV}} \approx 10^\circ$ .

The depth map  $d: \Omega \rightarrow \mathbb{R}_+$  associates a depth value  $d_{uv}$  to each pixel  $(u, v) \in \Omega$  in the canonical view. By inverting the camera model (5), we find that this corresponds to the 3D point  $P = d_{uv} \cdot K^{-1}p$ .

The viewpoint  $w \in \mathbb{R}^6$  represents an euclidean transformation  $(R, T) \in SE(3)$ , where  $w_{1:3}$  and  $w_{4:6}$  are rotation angles and translations along  $x, y$  and  $z$  axes respectively.

The map  $(R, T)$  transforms 3D points from the canonical view to the actual view. Thus a pixel  $(u, v)$  in the canonical view is mapped to the pixel  $(u', v')$  in the actual view by the warping function  $\eta_{d,w}: (u, v) \mapsto (u', v')$  given by

$$p' \propto K(d_{uv} \cdot RK^{-1}p + T), \quad (6)$$

where  $p' = (u', v', 1)$ .

Finally, the reprojection function  $\Pi$  takes as input the depth  $d$  and the viewpoint change  $w$  and applies the resulting warp to the canonical image  $\mathbf{J}$  to obtain the actual image  $\hat{\mathbf{I}} = \Pi(\mathbf{J}, d, w)$  as  $\hat{\mathbf{I}}_{u',v'} = \mathbf{J}_{uv}$ , where  $(u, v) = \eta_{d,w}^{-1}(u', v')$ . Note that this requires to compute the *inverse* of the warp  $\eta_{d,w}$ , which is detailed in Section 3.5.

The canonical image  $\mathbf{J} = \Lambda(a, d, l)$  is in turn generated as a combination of albedo, normal map and light direction. To do so, given the depth map  $d$ , we derive the normal map  $n: \Omega \rightarrow \mathbb{S}^2$  by associating to each pixel  $(u, v)$  a vector normal to the underlying 3D surface. In order to find this vector, we compute the vectors  $t_{uv}^u$  and  $t_{uv}^v$  tangent to the surface along the  $u$  and  $v$  directions. For example, the first one is

$$t_{uv}^u = d_{u+1,v} \cdot K^{-1}(p + e_x) - d_{u-1,v} \cdot K^{-1}(p - e_x), \quad (7)$$

where  $p$  is defined above and  $e_x = (1, 0, 0)$ . Then, the normal is obtained by taking the vector product  $n_{uv} \propto t_{uv}^u \times t_{uv}^v$ .

The normal  $n_{uv}$  is multiplied by the light direction  $l$  to obtain a value for the directional illumination and the latter is added to the ambient light. Finally, the result is multiplied by the albedo to obtain the illuminated texture, as follows:

$$\mathbf{J}_{uv} = (k_s + k_d \max\{0, \langle l, n_{uv} \rangle\}) \cdot a_{uv}. \quad (8)$$

Here  $k_s$  and  $k_d$  are the scalar coefficients weighting the ambient and diffuse terms, and are predicted by the model with range between 0 and 1 via rescaling a  $\tanh$  output. The light direction  $l = (l_x, l_y, 1)^T / (l_x^2 + l_y^2 + 1)^{0.5}$  is modeled as a spherical sector by predicting  $l_x$  and  $l_y$  with  $\tanh$ .

### 3.4 Perceptual Loss

The  $L_1$  loss function Eq. (3) is sensitive to small geometric imperfections and tends to result in blurry reconstructions. We add a *perceptual loss* term to mitigate this problem. The  $k$ th layer of an off-the-shelf image encoder  $e$  (VGG16 in our case [73]) predicts a representation  $e^{(k)}(\mathbf{I}) \in \mathbb{R}^{C_k \times W_k \times H_k}$  where  $\Omega_k = \{0, \dots, W_k - 1\} \times \{0, \dots, H_k - 1\}$  is the corresponding spatial domain. Note that this feature encoder does not have to be trained with supervised tasks. Self-supervised encoders can be equally effective as shown in Table 3.

Similar to Eq. (3), assuming a Gaussian distribution, the perceptual loss is given by

$$\mathcal{L}_p^{(k)}(\hat{\mathbf{I}}, \mathbf{I}, \sigma^{(k)}) = -\frac{1}{|\Omega_k|} \sum_{uv \in \Omega_k} \ln \frac{1}{\sqrt{2\pi(\sigma_{uv}^{(k)})^2}} \exp - \frac{(\ell_{uv}^{(k)})^2}{2(\sigma_{uv}^{(k)})^2}, \quad (9)$$

where  $\ell_{uv}^{(k)} = |e_{uv}^{(k)}(\hat{\mathbf{I}}) - e_{uv}^{(k)}(\mathbf{I})|$  for each pixel index  $uv$  in the  $k$ th layer. We also compute the loss for  $\hat{\mathbf{I}}'$  using  $\sigma^{(k)'} \cdot \sigma^{(k)}$  and  $\sigma^{(k)'}$  are additional confidence maps predicted by our model. In practice, we found it is good enough for our purpose to use the features from only one layer (relu3\_3) of VGG16. We therefore shorten the notation of perceptual loss to  $\mathcal{L}_p$ . With this, the loss function  $\mathcal{L}$  in Eq. (4) is replaced by  $\mathcal{L} + \lambda_p \mathcal{L}_p$  with  $\lambda_p = 1$ .

### 3.5 Differentiable Rendering Layer

As noted in Section 3.3, the reprojection function  $\Pi$  *warp*s the canonical image  $\mathbf{J}$  to generate the actual image  $\mathbf{I}$ . In CNNs, image warping is usually regarded as a simple operation that can be implemented efficiently using a bilinear resampling layer [74]. However, this is true only if we can easily send pixels  $(u', v')$  in the warped image  $\mathbf{I}$  back to pixels  $(u, v)$  in the source image  $\mathbf{J}$ , a process also known as *backward warping*. Unfortunately, in our case the function  $\eta_{d,w}$  obtained by Eq. (6) sends pixels the opposite way.

Implementing a *forward warping* layer is surprisingly delicate. One way of approaching the problem is to regard this task as a special case of rendering a textured mesh. The *Neural Mesh Renderer* (NMR) of [68] is a differentiable renderer of this type. In our case, the mesh has one vertex per pixel and each group of  $2 \times 2$  adjacent pixels is tessellated by two triangles. Empirically, we found the quality of the texture gradients of NMR to be poor in this case, likely caused by noisy depth map  $d$  and high frequency content in the texture image  $\mathbf{J}$ .

We solve the problem as follows. First, we use NMR to warp only the depth map  $d$ , obtaining a version  $\bar{d}$  of the depth map as seen from the input viewpoint. This has two advantages: backpropagation through NMR is faster and second, the depth gradients are more stable than color gradients, probably also due to the comparatively smooth nature of the depth map  $d$  compared to the texture image  $\mathbf{J}$ .

Given the depth map  $\bar{d}$ , we then use the inverse of Eq. (6) to find the warp field from the observed viewpoint to the canonical viewpoint, and bilinearly resample the canonical image  $\mathbf{J}$  to obtain the reconstruction (i.e. using backward warping).

## 4 EXPERIMENTS

In this section, we first describe the experimental setup and implementation details, and then present the qualitative results on three object categories, human faces, cat faces and synthetic cars, followed by extensive ablation studies and analyses. We also report comparisons with several state-of-the-art methods both qualitatively and quantitatively. In the end, we provide a discussion on the limitations of our method.

### 4.1 Experimental Setup

#### 4.1.1 Datasets

We test our method on three human face datasets: *CelebA* [75], *3DFAW* [76], [77], [78], [79] and *BFM* [11]. CelebA is a large scale human face dataset, consisting of over 200k images of real human faces in the wild annotated with bounding boxes. 3DFAW contains 23k images with 66 3D keypoint annotations, which we use to evaluate our 3D predictions in Section 4.4. We roughly crop the images around the head region using MTCNN [80] and use the official train/val/test splits. BFM (Basel Face Model) is a synthetic face model, which we use to assess the quality of the 3D reconstructions (since the in-the-wild datasets lack ground-truth). We follow the protocol of [19] to generate a dataset, sampling shapes, poses, textures, and illumination randomly. We use images from SUN Database [81] as background and save ground truth depth maps for evaluation.

We also test our method on cat faces and synthetic cars. We use two *cat datasets* [82], [83]. The first one has 10k cat images with nine keypoint annotations, and the second one is a collection of dog and cat images, containing 1.2k cat images with bounding box annotations. We combine the two datasets and crop the images around the cat heads. For cars, we render 35k images of synthetic cars from *ShapeNet* [84] with random viewpoints and illumination. We randomly split the images by 8:1:1 into train, validation and test sets.

#### 4.1.2 Metrics

Since the scale of 3D reconstruction from projective cameras is inherently ambiguous [35], we discount it in the evaluation. Specifically, given the depth map  $d$  predicted by our model in the canonical view, we warp it to a depth map  $\bar{d}$  in the actual view using the predicted viewpoint and compare the latter to the ground-truth depth map  $d^*$  using the *scale-invariant depth error* (SIDE) [85]

$$E_{\text{SIDE}}(\bar{d}, d^*) = \left( \frac{1}{WH} \sum_{uv} \Delta_{uv}^2 - \left( \frac{1}{WH} \sum_{uv} \Delta_{uv} \right)^2 \right)^{\frac{1}{2}}, \quad (10)$$

where  $\Delta_{uv} = \log \bar{d}_{uv} - \log d_{uv}^*$ . We compare only valid depth pixels and erode the foreground mask by one pixel to discount rendering artefacts at object boundaries. Additionally, we report the *mean angle deviation* (MAD) between normals

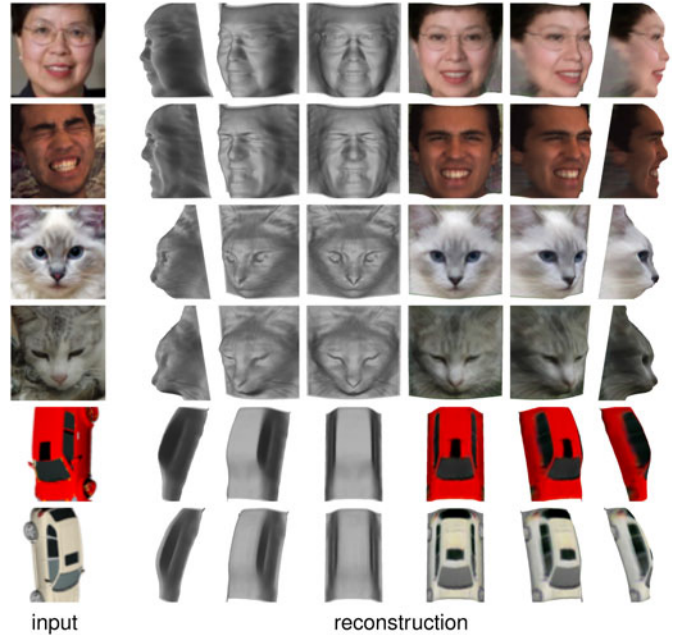


Fig. 3. *Reconstruction of faces, cats and cars.* Our unsupervised model recovers accurate 3D shape from only a single input image.

computed from ground truth depth and from the predicted depth, measuring how well the surface is captured.

#### 4.1.3 Implementation Details

The function  $(d, a, w, l, \sigma) = \Phi(\mathbf{I})$  that predicts depth, albedo, viewpoint, lighting, and confidence maps from the image  $\mathbf{I}$  is implemented using individual neural networks. The depth and albedo are generated by encoder-decoder networks, while viewpoint and lighting are regressed using simple encoder networks. The encoder-decoders do not use skip connections because input and output images are *not* spatially aligned (since the output is in the canonical viewpoint). All four confidence maps are predicted using the same network, at different decoding layers for the photometric and perceptual losses since these are computed at different resolutions. The final activation function is  $\tanh$  for depth, albedo, viewpoint and lighting and  $\text{softplus}$  for the confidence maps. The depth prediction is centered on the mean before  $\tanh$ , as the global distance is estimated as part of the viewpoint. We do *not* use any special initialization for all predictions, except that two border pixels of the depth maps on both the left and the right are clamped at a maximal depth to avoid boundary issues.

We train using Adam over batches of 64 input images, resized to  $64 \times 64$  pixels. The size of the output depth and albedo is also  $64 \times 64$ . We train for approximately 50k iterations. For visualization, depth maps are upsampled to 256. We include more details in the supplementary material, which can be found on the Computer Society Digital Library at <http://doi.ieeecomputersociety.org/10.1109/TPAMI.2021.3076536>.

## 4.2 Qualitative Results

### 4.2.1 Reconstruction Results

In Fig. 3 we show reconstruction results of human faces from CelebA and 3DFAW, cat faces from [82], [83] and synthetic cars from ShapeNet. The 3D shapes are recovered

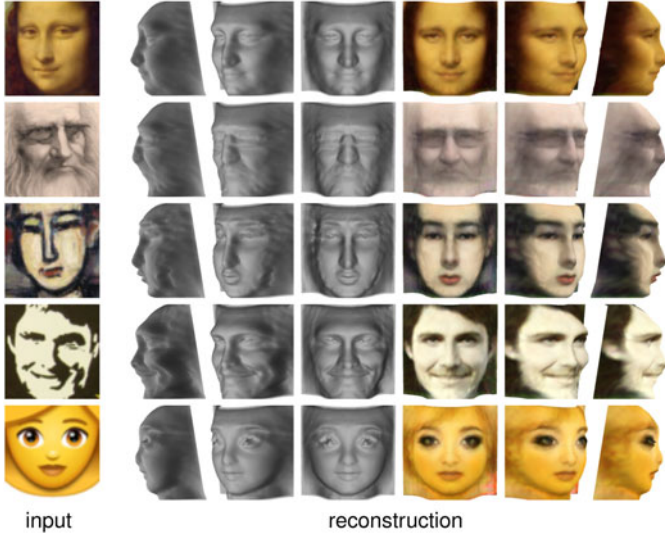


Fig. 4. *Reconstruction of faces in paintings and cartoons.* The model trained on real faces in CelebA generalizes well to abstract faces in paintings and cartoons.



Fig. 6. *Frame-by-frame reconstruction on video sequences.* Even though our model does not use videos for training, it produces temporally consistent reconstructions on video sequences.

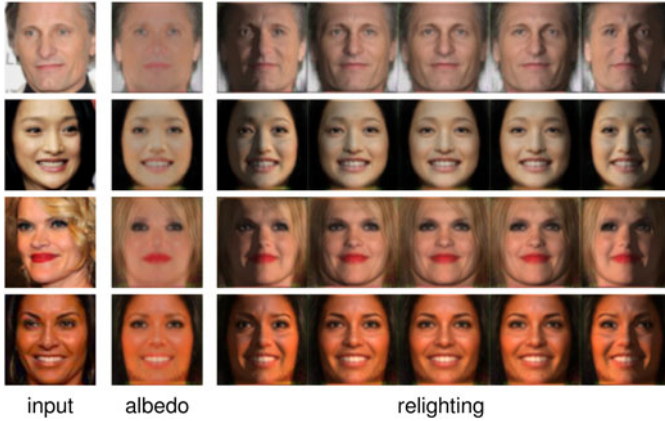


Fig. 5. *Re-lighting results.* Our model disentangles albedo and shading from a single input image, which allows us to relight the objects with novel lighting conditions.

with high fidelity. The reconstructed 3D face, for instance, contain fine details of the nose, eyes and mouth even in the presence of extreme facial expression.

#### 4.2.2 Generalization to Paintings

To further test generalization, we applied our model trained on the CelebA dataset to a number of paintings and cartoon drawings of faces collected from [86] and the Internet. As shown in Fig. 4, our method still works well even though it has never seen such images during training. It is worth noting that since the model is trained using real face images, the reconstructions seem to also be more “realistic” faces reflecting the prior learned during training.

#### 4.2.3 Relighting

A by-product of our reconstruction framework is that it learns to disentangle albedo and shading from a single image, without any external supervision at all. This is possible by leveraging the symmetry assumption on the albedo as well as the categorical prior imposed by training set.

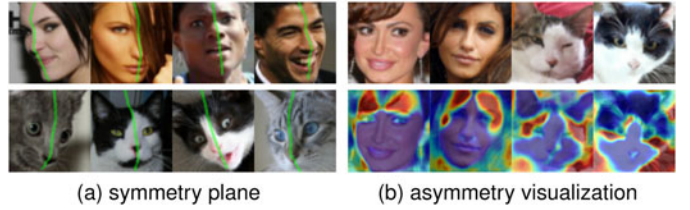


Fig. 7. *Symmetry plane and asymmetry detection.* (a): our model can reconstruct the “intrinsic” symmetry plane of an in-the-wild object even though the appearance is highly asymmetric. (b): asymmetries (highlighted in red) are detected and visualized using confidence map  $\sigma'$ .

Decomposing the albedo map enables realistic graphics editing applications, such as re-rendering the object under different lighting conditions, as illustrated in Fig. 5.

#### 4.2.4 Inference on Video Frames

We can also apply our trained model on video sequences frame-by-frame. To demonstrate this, we take speech clips from VoxCeleb [87], and crop the faces using MTCNN<sup>3</sup> [80]. We then feed the crops to our model to produce a 3D reconstruction of the faces and render them from novel viewpoints, shown in Fig. 6. Note that our model does not use videos for training, yet it produces temporally consistent and accurate reconstruction results by simply processing the frames independently.

#### 4.2.5 Symmetry and Asymmetry Detection

Since our model predicts a canonical view of the objects that is symmetric about the vertical center-line of the image, we can easily visualize the symmetry plane, which is otherwise non-trivial to detect from in-the-wild images. In Fig. 7, we warp the center-line of the canonical image to the predicted input viewpoint. Our method can detect symmetry planes accurately despite the presence of asymmetric texture and lighting

3. We use the implementation from <https://github.com/timesler/facenet-pytorch>.

TABLE 2  
Comparison With Baselines

No	Baseline	SIDE ( $\times 10^{-2}$ ) ↓	MAD (deg.) ↓
(1)	Supervised	$0.410 \pm 0.103$	$10.78 \pm 1.01$
(2)	Const. null depth	$2.723 \pm 0.371$	$43.34 \pm 2.25$
(3)	Average g.t. depth	$1.990 \pm 0.556$	$23.26 \pm 2.85$
(4)	Ours (unsupervised)	$0.793 \pm 0.140$	$16.51 \pm 1.56$

SIDE and MAD errors of our reconstructions on the BFM dataset compared against a fully-supervised and trivial baselines.

TABLE 3  
Ablation Study

No	Method	SIDE ( $\times 10^{-2}$ ) ↓	MAD (deg.) ↓
(1)	Ours full	$0.793 \pm 0.140$	$16.51 \pm 1.56$
(2)	w/o albedo flip	$2.916 \pm 0.300$	$39.04 \pm 1.80$
(3)	w/o depth flip	$1.139 \pm 0.244$	$27.06 \pm 2.33$
(4)	w/o light	$2.406 \pm 0.676$	$41.64 \pm 8.48$
(5)	w/o perc. loss	$0.931 \pm 0.269$	$17.90 \pm 2.31$
(6)	w/ self-sup. perc. loss	$0.815 \pm 0.145$	$15.88 \pm 1.57$
(7)	w/o confidence	$0.829 \pm 0.213$	$16.39 \pm 2.12$

Refer to Section 4.3.2 for details.

effects. We also overlay the predicted confidence map  $\sigma'$  onto the image, confirming that the model assigns low confidence to asymmetric regions in a sample-specific way.

### 4.3 Analyses and Discussions

#### 4.3.1 Comparison With Baselines

Table 2 uses the BFM dataset to compare the depth reconstruction quality obtained by our method, a fully-supervised baseline and two other baselines. The supervised baseline is a version of our model trained to regress the ground-truth depth maps using an  $L_1$  loss. The trivial baseline predicts a constant uniform depth map, which provides a performance lower-bound. The third baseline is a constant depth map obtained by averaging all ground-truth depth maps in the test set. Our method largely outperforms the two constant baselines and approaches the results of supervised training.

#### 4.3.2 Ablation

To understand the influence of the individual parts of the model, we remove them one at a time and evaluate the performance of the ablated model in Table 3 and Fig. 8.

In the table, *row (1)* shows the performance of the full model (the same as in Table 2). *Row (2)* does not flip the albedo. Thus, the albedo is not encouraged to be symmetric in the canonical space, which fails to canonicalize the viewpoint of the object and to use cues from symmetry to recover shape. The performance is as low as the trivial baseline in Table 2. *Row (3)* does not flip the depth, with a similar effect to row (2). In addition, we had to add an  $L_2$  smoothness loss on the depth maps during training. Otherwise, the model tends to produce noisy depth maps without the symmetry constraint, which lead to heavy occlusion and break the training.

*Row (4)* predicts a shading map instead of computing it from depth and light direction. This also harms performance

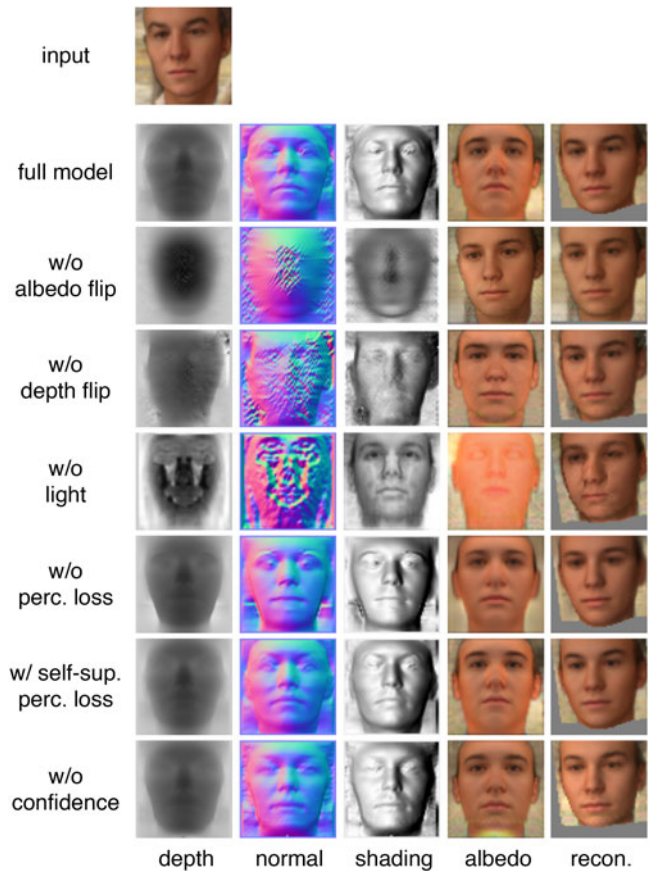


Fig. 8. Ablation study. Refer to Section 4.3.2 for details.

significantly because shading cannot be used as a cue to recover shape. Moreover, the training often collapses after a few epochs as the model produces spikes in the depth maps that also result in large occlusion. We therefore report the results of the latest epoch prior to collapse.

*Row (5)* switches off the perceptual loss, which leads to degraded image quality and hence degraded reconstruction results. *Row (6)* replaces the ImageNet pretrained image encoder used in the perceptual loss with one<sup>4</sup> trained through a self-supervised task [88], which shows no difference in performance.

Finally, *row (7)* switches off the confidence maps, using a fixed and uniform value for the confidence — this reduces losses (3) and (9) to the basic  $L_1$  and  $L_2$  losses, respectively. The accuracy does not drop significantly, as faces in BFM are highly symmetric (e.g. do not have hair), but its variance increases. To better understand the effect of the confidence maps, we specifically evaluate on partially asymmetric faces using perturbations.

#### 4.3.3 Asymmetric Perturbation

In order to demonstrate that our uncertainty modelling allows the model to handle asymmetry, we add asymmetric perturbations to BFM. Specifically, we generate random rectangular color patches with 20 to 50 percent of the image size and blend them onto the images with  $\alpha$ -values ranging from 0.5 to 1, as shown in Fig. 9. We then train our model

4. We use a RotNet [88] pretrained VGG16 model obtained from <https://github.com/facebookresearch/DeeperCluster>.

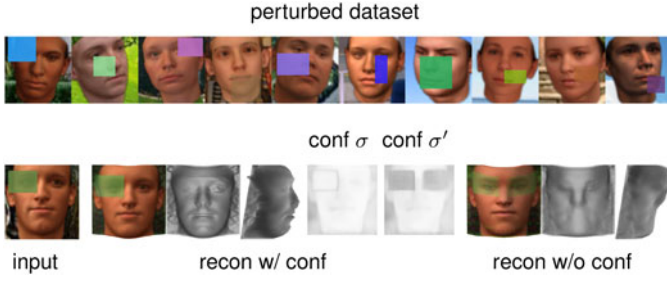


Fig. 9. *Asymmetric perturbation.* Top: examples of the perturbed dataset. Bottom: reconstructions with and without confidence maps. Confidence allows the model to correctly reconstruct the 3D shape with the asymmetric texture.

TABLE 4  
Asymmetric Perturbation

	SIDE ( $\times 10^{-2}$ ) $\downarrow$	MAD (deg.) $\downarrow$
No perturb, no conf.	$0.829 \pm 0.213$	$16.39 \pm 2.12$
No perturb, conf.	$0.793 \pm 0.140$	$16.51 \pm 1.56$
Perturb, no conf.	$2.141 \pm 0.842$	$26.61 \pm 5.39$
Perturb, conf.	$0.878 \pm 0.169$	$17.14 \pm 1.90$

We add asymmetric perturbations to BFM and show that confidence maps allow the model to reject such noise, while the vanilla model without confidence maps breaks.

with and without confidence on these perturbed images, and report the results in Table 4. Without the confidence maps, the model always predicts a symmetric albedo and geometry reconstruction often fails. With our confidence estimates, the model is able to reconstruct the asymmetric faces correctly, with very little loss in accuracy compared to the unperturbed case.

#### 4.3.4 Training Only on Frontal Faces

Our full training data consists of single-view images of many instances, each captured from a different viewpoint, which essentially compose a large “multi-view” image set, although these are “multi-views” of different instances with different texture and shape. Nonetheless, it would be interesting to know how much this “multi-view” signal contributes to the learning, compared to other cues, such as symmetry and shading.

In order to understand this, we generate another synthetic face dataset consisting of only frontal faces with random texture and shape variations, and train a model on only frontal faces. We compare the performance of this model to our full model trained on the original dataset of images with various viewpoints in Table 5 and Fig. 10. In fact, the model trained on only frontal faces is indeed able to learn 3D shape of frontal faces, despite producing artifacts and a lower reconstruction accuracy compared to the full model. This suggests the symmetry and shading constraints can still provide powerful signals for learning shapes, even without the view variation in the training set. However, this model fails to generalize to input faces from other viewpoints.

#### 4.3.5 Training With Fewer Images

As the symmetry assumption and shading seem to provide strong signals for learning the shape, another interesting

TABLE 5  
Training on Frontal Faces

	SIDE ( $\times 10^{-2}$ ) $\downarrow$	MAD (deg.) $\downarrow$
Train frontal, test frontal	$1.347 \pm 0.150$	$22.90 \pm 1.13$
Train frontal, test all	$1.858 \pm 0.429$	$30.80 \pm 3.93$
Train all, test frontal	$0.818 \pm 0.107$	$15.90 \pm 1.22$
Train all, test all	$0.793 \pm 0.140$	$16.51 \pm 1.56$

We compare the model trained on only frontal faces with our full model trained on faces with all random viewpoints.

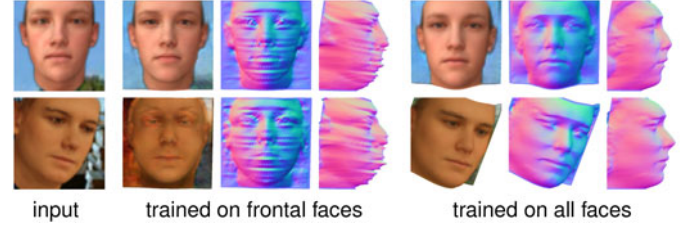


Fig. 10. *Training only on frontal faces.* The model trained on only frontal faces is still able to learn 3D shape, despite producing artifacts (first row), but it does not generalize to other views (second row).

question to ask is: does it still need to be trained on a large image collection? To answer this question, we train the model on different numbers of training images, ranging from only one single image to the entire training set of 155k images, and compare the results in Fig. 11. When training with 1 image and 100 images, we added a L2 smoothness loss on the depth maps, as the training otherwise collapses due to noisy depth maps.

As shown in Fig. 11, when trained on only 1 image, the model seems still able to pick up some shading and symmetry cues to recover the 3D shape. However, these cues alone cannot provide enough constraints on this heavily ill-posed 2D-to-3D task. Therefore, although the image reconstruction loss is low, the underlying 3D shape is poorly reconstructed. The model only starts to learn reasonable 3D faces when trained on 1000 or more images, which suggests that a sufficiently large image collection is critical for the model to learn a 3D shape prior of the object category.

#### 4.3.6 Mixing Categories

In order to understand whether the model learns different priors for different categories, we further conduct experiments on cross-category inference as well as multi-category training. Fig. 12 shows some examples. We first feed images of human faces to a model trained on images of cat images and also the other way around. Unsurprisingly, the models trained on one single category learn shape priors specific to that particular category, and tend to reconstruct shapes of the training category, even if the input images depict a different category.

We further consider training the model on a mixture of images from two object categories, which turns out still capable of reconstructing both categories with similar quality compared to the models trained individually on each category. This observation shows promise of learning a general modal independent of object categories in the future.

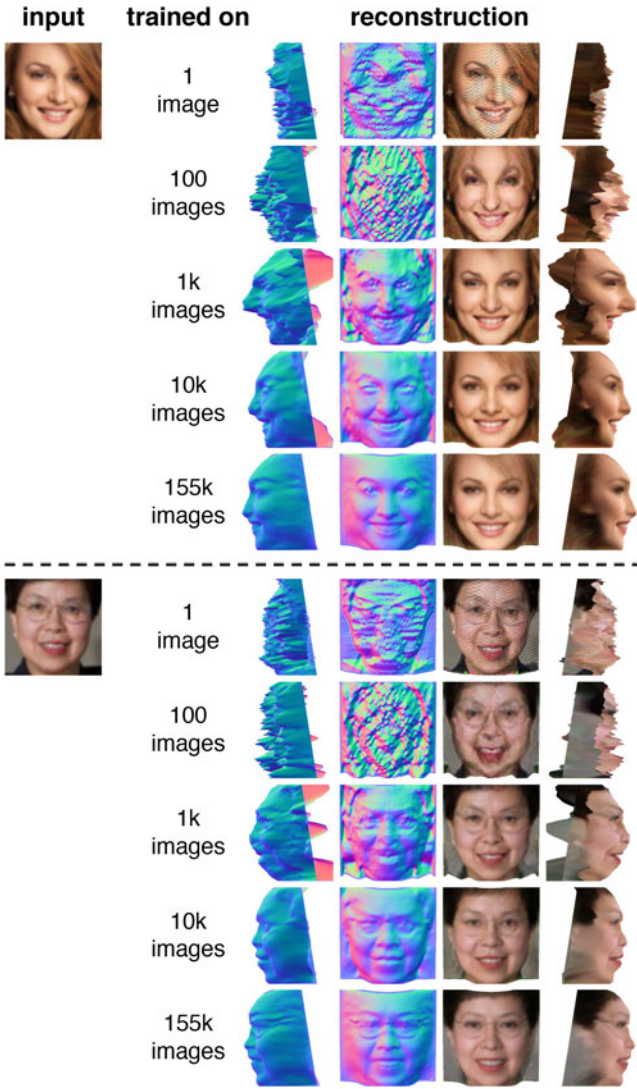


Fig. 11. *Training with fewer images.* We show a qualitative comparison of the models trained with different numbers of images, which confirms the necessity of training on a sufficiently large image collection.

#### 4.4 Comparison With the State of the Art

As shown in Table 1, most reconstruction methods in the literature require either image annotations, prior 3D models or both. When these assumptions are dropped, the task becomes considerably harder, and there is little prior work that is directly comparable. Of these, [33] only uses synthetic, texture-less objects from ShapeNet, [8] reconstructs in-the-wild faces but does not report any quantitative results, and [7] reports quantitative results only on keypoint regression, but not on the 3D reconstruction quality. We were not able to obtain code or trained models from [7], [8] for a direct quantitative comparison and thus compare qualitatively.

##### 4.4.1 Qualitative Comparison

In order to establish a side-by-side comparison, we cropped the figures reported in the papers [7], [8] and compare our results with theirs (Fig. 13). Our method produces higher quality reconstructions than both methods, with fine details of the facial expression. The difference is especially noticeable in the recovery of 3D shape for [7], and the shape

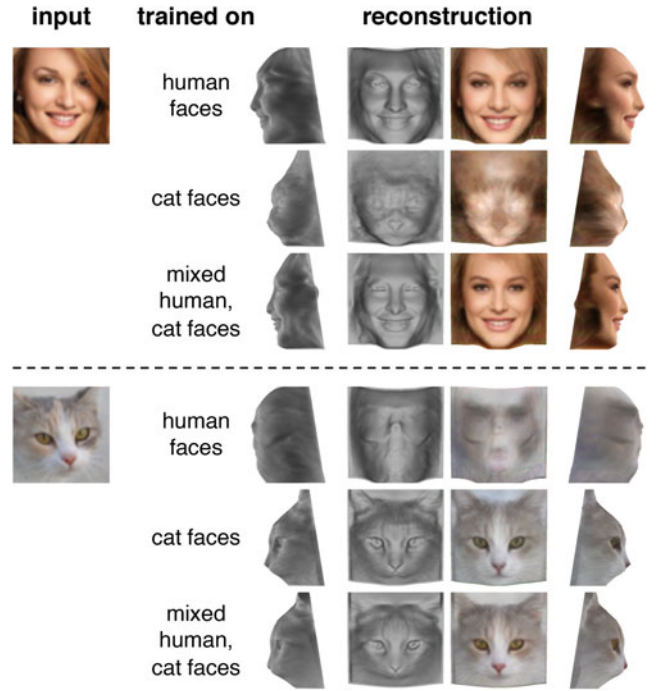


Fig. 12. *Mixing categories.* When trained on one single category, the model learns a prior specific to that particular category, whereas when trained on two categories, it is able to reconstruct both categories well.

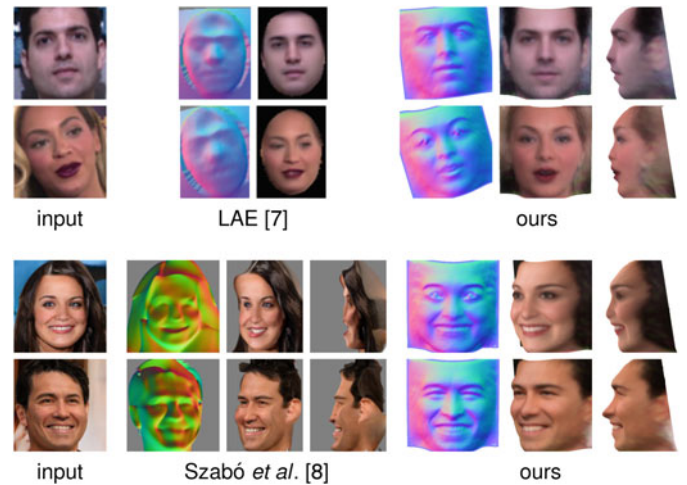


Fig. 13. *Qualitative comparison to SOTA.* Comparing to [7], [8], our method recovers higher quality shapes.

generation in [8]. Note that [8] uses an unconditional GAN that *generates* high resolution 3D faces from random noise, and cannot *recover* 3D shapes from images. The input images for [8] in Fig. 13 were generated by their GAN.

##### 4.4.2 3D Keypoint Depth Evaluation

Next, we compare to the DepthNet model of [9]. This method predicts depth for selected facial keypoints, but uses 2D keypoint annotations as input — a much easier setup than the one we consider here. Still, we compare the quality of the reconstruction of these sparse point obtained by DepthNet and our method. We also compare to the baselines MOFA [90] and AIGN [89] reported in [9]. For a fair comparison, we use their public code which computes the

TABLE 6  
3DFAW Keypoint Depth Evaluation

	Depth Corr. $\uparrow$
Ground truth	66
AIGN [89] ( <b>supervised</b> , from [9])	50.81
DepthNetGAN [9] ( <b>supervised</b> , from [9])	58.68
MOFA [90] ( <b>model-based</b> , from [9])	15.97
DepthNet [9] (from [9])	26.32
DepthNet [9] (from GitHub)	35.77
Ours	48.98
Ours (w/ CelebA pre-training)	54.65

Depth correlation between ground truth and prediction evaluated at 66 facial keypoint locations.

depth correlation score (between 0 and 66) on the frontal faces. We use the 2D keypoint locations to sample our predicted depth and then evaluate the same metric. The set of test images from 3DFAW and the preprocessing are identical to [9]. Since 3DFAW is a small dataset with limited variation, we also report results with CelebA pre-training.

In Table 6 we report the results from their paper and the slightly improved results we obtained from their publicly-available implementation. The paper also evaluates a supervised model using a GAN discriminator trained with ground-truth depth information. While our method does not use any supervision, it still outperforms DepthNet and reaches close-to-supervised performance.

#### 4.4.3 3D Face Reconstruction Benchmarks

We also evaluate the reconstructed 3D meshes and compare the performance with several recent 3DMM-based reconstruction methods [21], [57], [58], [59], [60], [61] on two 3D face reconstruction benchmarks [21], [91].

The first benchmark by Feng *et al.* [91] provides a test set, which consists of 133 ground-truth 3D scans and 2,000 test images, including 656 high-quality (HQ) images captured in a controlled environment and 1,344 low-quality (LQ) images extracted from videos. The second one, NoW benchmark [21], provides a test set of 1,702 images of 80 subjects and a ground-truth 3D scan per subject. These images are captured with a higher variety in facial expression, occlusion, and lighting, compared to the Feng *et al.* benchmark.

However, it is important to highlight that these benchmarks are designed specifically for evaluating 3DMM-based face reconstruction methods, and inherently put model-free approaches at a disadvantage. In both of these benchmark sets, only 3D scans of neutral faces are available, which are used as ground-truth for various input images that describe different viewpoints and facial expressions and may contain occlusion. This gives the 3DMM-based methods an advantage over our method, since the output of these methods is always constrained to a face model regardless of input variety, whereas our method produces instance-specific reconstructions with different expressions, which are not captured in the ground-truth scans. Our main intention with this evaluation is the establishment of a fair, quantitative evaluation of future model-free methods, since qualitative comparisons are often subjective and synthetic benchmarks are limited in terms of generalization to real data.

TABLE 7  
Performance on Feng *et al.* [91] Benchmark

Methods	Median $\downarrow$		Mean $\downarrow$		Std	
	LQ	HQ	LQ	HQ	LQ	HQ
Extreme3D [58]	2.40	2.37	3.49	3.58	6.15	6.75
3DMM-CNN [57]	1.88	1.85	2.32	2.29	1.89	1.88
PRNet [59]	1.79	1.60	2.38	2.06	2.19	1.79
RingNet [21]	1.63	1.58	2.08	2.02	1.79	1.69
3DDFA-V2 [60]	1.62	1.49	2.10	1.91	1.87	1.64
DECA [61]	1.48	1.44	1.91	1.89	1.68	1.66
Const. flat plane	12.47	12.47	14.11	14.07	10.21	10.17
Ours ( <b>model-free</b> )	5.58	5.54	5.74	5.68	1.47	1.89

We compare our model-free unsupervised method with several recent 3DMM-based methods.

For both datasets, we detect faces and crop the images using MTCNN [80] and obtain 3D mesh reconstructions from the depthmaps predicted by our model trained on CelebA. We then use the same evaluation protocol in both benchmarks [21], [91], which align the predicted meshes with the ground-truth meshes with a rigid transformation based on 7 pre-defined keypoints and compute the scan-to-mesh distances. We obtain these keypoints on our predicted meshes by applying a facial keypoint detector [92] on the reconstructed canonical images. The average keypoints are used when the keypoint detector fails.

We report the statistics of the distances and compare them with other methods in Tables 7 and 8. Although our model-free unsupervised method does not perform as well as the model-based methods on these benchmarks, it is significantly better than a flat shape baseline as shown in Table 7. Since the NoW dataset provides attributes for the images, we select a subset of the test set that contains 91 frontal neutral faces, which better match with the ground-truth scans, and include the results in Table 8. The results in this subset further reduce the gap towards model-based methods.

#### 4.5 Limitations

While our unsupervised method is robust in many challenging scenarios (e.g., extreme facial expression, drawings), we do observe limitations as shown in Fig. 14.

TABLE 8  
Performance on NoW *et al.* [21] Benchmark

Methods	Median $\downarrow$	Mean $\downarrow$	Std
3DMM-CNN [57]	1.84	2.33	2.05
PRNet [59]	1.50	1.98	1.88
RingNet [21]	1.21	1.54	1.31
3DDFA-V2 [60]	1.23	1.57	1.39
DECA [61]	1.09	1.38	1.18
Ours ( <b>model-free</b> )	2.64	3.29	2.86
3DMM-CNN [57] (frontal)	1.88	2.36	2.07
PRNet [59] (frontal)	1.38	1.79	1.67
RingNet [21] (frontal)	1.16	1.48	1.28
Ours ( <b>model-free</b> , frontal)	2.25	2.80	2.44

We compare our model-free unsupervised method with several recent 3DMM-based methods. The bottom half of the table reports the results on a subset of frontal neutral faces, indicated by “frontal”.

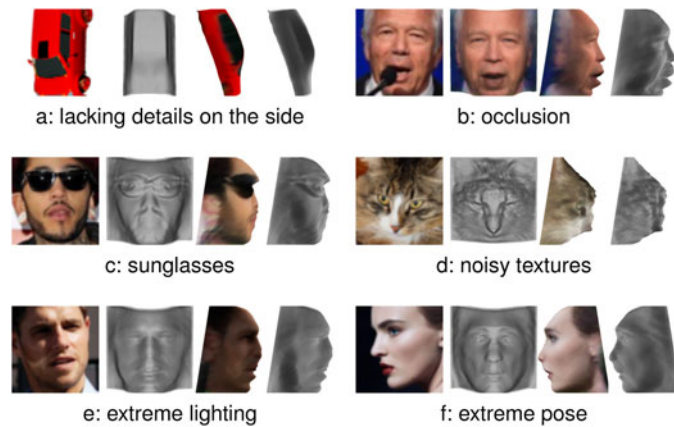


Fig. 14. *Limitations*. See Section 4.5 for details.

First and foremost, our model relies on the assumption that object category has weakly symmetric 3D shape as well as weakly symmetric albedo. Extending the key insights in this work, including leveraging category priors, other forms of symmetry and shape from shading in a learning framework, to general objects will require future work.

In this work, we represent shape using a depth map in the canonical (symmetric) viewpoint, which cannot describe the full 3D shape in 360 degrees. Thus, the reconstructed shapes often lack details on the sides. This is particularly evident for the cars, as illustrated in Fig. 14a. One would need to consider using other 3D representations to capture full 3D objects from 360 degrees.

Our model also tends to ignore occluders (Fig. 14b), since the training set does not contain many examples with occlusion. Disentangling dark textures and shading is often difficult. Therefore, the model fails to accurately reconstruct sunglasses (Fig. 14c) and may produce bumpy surfaces when the texture is noisy (Fig. 14d). During training, we assume a simple Lambertian shading model, ignoring shadows and specularities, which leads to inaccurate reconstructions under extreme lighting conditions (Fig. 14e) or highly non-Lambertian surfaces. The reconstruction quality is also lower for extreme poses (Fig. 14f), partly due to poor supervisory signal from the reconstruction loss of side images. This may be improved by imposing constraints from accurate reconstructions of frontal poses.

## 5 CONCLUSION

We have presented a method that can learn a 3D model of a deformable object category from an unconstrained collection of single-view images of the object category. The model is able to obtain high-fidelity monocular 3D reconstructions of individual object instances. This is trained based on a reconstruction loss without any supervision, resembling an autoencoder. We have shown that symmetry and illumination are strong cues for shape and help the model to converge to a meaningful reconstruction. Our model outperforms a current state-of-the-art 3D reconstruction method that uses 2D keypoint supervision. As for future work, the model currently represents 3D shape from a canonical viewpoint using a depth map, which is sufficient for objects such as faces that have a roughly convex shape and a natural canonical viewpoint. For more complex

objects, it may be possible to extend the model to use either multiple canonical views or a different 3D representation, such as a mesh or a voxel map.

## ACKNOWLEDGMENTS

The authors would like to thank Soumyadip Sengupta for sharing with us the code to generate synthetic face datasets and Mihir Sahasrabudhe for sending us the reconstruction results of Lifting AutoEncoders. The authors would also like to thank the members of Visual Geometry Group for insightful discussions. This work was supported in part by the Facebook Research, in part by the ERC Horizon 2020 Research, and in part by the Innovation Programme under Grant IDIU 638009.

## REFERENCES

- [1] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness," in *Proc. Int. Conf. Learn. Representations*, 2019. [Online]. Available: <https://openreview.net/forum?id=Bygh9j09KX>
- [2] D. P. Mukherjee, A. Zisserman, and J. M. Brady, "Shape from symmetry: Detecting and exploiting symmetry in affine images," *Philos. Trans. Roy. Soc. London*, vol. 351, pp. 77–106, 1995.
- [3] A. R. J. François, G. G. Medioni, and R. Waupotitsch, "Mirror symmetry  $\Rightarrow$  2-view stereo geometry," *Image Vis. Comput.*, vol. 21, pp. 137–143, 2003.
- [4] S. Thrun and B. Wegbreit, "Shape from symmetry," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1824–1831.
- [5] S. N. Sinha, K. Ramnath, and R. Szeliski, "Detecting and reconstructing 3D mirror symmetric objects," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 586–600.
- [6] Y. Gao and A. L. Yuille, "Exploiting symmetry and/or manhattan properties for 3D object structure estimation from single and multiple images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7408–7417.
- [7] M. Sahasrabudhe, Z. Shu, E. Bartrum, R. A. Guler, D. Samaras, and I. Kokkinos, "Lifting autoencoders: Unsupervised learning of a fully-disentangled 3D morphable model using deep non-rigid structure from motion," in *Proc. Int. Conf. Comput. Vis. Workshops*, 2019, pp. 4054–4064.
- [8] A. Szabó, G. Meishvili, and P. Favaro, "Unsupervised generative 3D shape learning from natural images," 2019, *arXiv:1910.00287*.
- [9] J. R. A. Moniz, C. Beckham, S. Rajotte, S. Honari, and C. Pal, "Unsupervised depth estimation, 3D face rotation and replacement," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 9759–9769.
- [10] S. Wu, C. Rupprecht, and A. Vedaldi, "Unsupervised learning of probably symmetric deformable 3D objects from images in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 1–10.
- [11] P. Paysan, R. Knothe, B. Amberg, S. Romdhani, and T. Vetter, "A 3D face model for pose and illumination invariant face recognition," in *Proc. IEEE Int. Conf. Adv. Video Signal Based Surveill.*, 2009, pp. 296–301.
- [12] J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling," in *Proc. Neural Inf. Process. Syst.*, 2016, pp. 82–90.
- [13] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 40–49.
- [14] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, "Generating 3D faces using convolutional mesh autoencoders," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 725–741.
- [15] Z. Geng, C. Cao, and S. Tulyakov, "3D guided fine-grained face manipulation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9813–9822.
- [16] B. Gecer, S. Ploumpis, I. Kotsia, and S. Zafeiriou, "GANFIT: Generative adversarial network fitting for high fidelity 3D face reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 1155–1164.

- [17] T. Gerig *et al.*, "Morphable face models - An open framework," in *Proc. Int. Conf. Autom. Face Gesture Recognit.*, 2018, pp. 75–82.
- [18] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, "End-to-end recovery of human shape and pose," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7122–7131.
- [19] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. W. Jacobs, "SfSNet: Learning shape, reflectance and illuminance of faces in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6296–6305.
- [20] M. Wang, Z. Shu, S. Cheng, Y. Panagakis, D. Samaras, and S. Zafeiriou, "An adversarial neuro-tensorial approach for learning disentangled representations," *Int. J. Comput. Vis.*, vol. 127, no. 6–7, pp. 743–762, 2019.
- [21] S. Sanyal, T. Bolkart, H. Feng, and M. Black, "Learning to regress 3D face shape and expression from an image without 3D supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 7763–7772.
- [22] M. Gadelha, S. Maji, and R. Wang, "3D shape induction from 2D views of multiple objects," in *Proc. Int. Conf. 3D Vis.*, 2017, pp. 402–411.
- [23] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3D object reconstruction without 3D supervision," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 1704–1712.
- [24] H. Kato and T. Harada, "Learning view priors for single-view 3D reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 9778–9787.
- [25] A. Kanazawa, S. Tulsiani, A. A. Efros, and J. Malik, "Learning category-specific mesh reconstruction from image collections," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 371–386.
- [26] W. Chen *et al.*, "Learning to predict 3D objects with an interpolation-based differentiable renderer," in *Proc. Conf. Neural Inf. Process. Syst.*, 2019, pp. 9605–9616. [Online]. Available: <https://dblp.org/rec/conf/nips/ChenLGSLJF19.html?view=bibtex>
- [27] P. Henzler, N. Mitra, and T. Ritschel, "Escaping plato's cave using adversarial training: 3D shape from unstructured 2D image collections," in *Proc. Int. Conf. Comput. Vis.*, 2019, pp. 9984–9993.
- [28] N. Kulkarni, A. Gupta, D. F. Fouhey, and S. Tulsiani, "Articulation-aware canonical surface mapping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 452–461.
- [29] S. Goel, A. Kanazawa, and J. Malik, "Shape and viewpoints without keypoints," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 88–104.
- [30] X. Li *et al.*, "Self-supervised single-view 3D reconstruction via semantic consistency," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 677–693.
- [31] T. Nguyen-Phuoc, C. Li, L. Theis, C. Richardt, and Y.-L. Yang, "Hologan: Unsupervised learning of 3D representations from natural images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7588–7597.
- [32] K. Schwarz, Y. Liao, M. Niemeyer, and A. Geiger, "GRAF: Generative radiance fields for 3D-aware image synthesis," in *Proc. Conf. Neural Inf. Process. Syst.*, 2020, pp. 20154–20166. [Online]. Available: <https://papers.nips.cc/paper/2020/hash/e92e1b476bb5262d793fd40931e0ed53-Abstract.html>
- [33] P. Henderson and V. Ferrari, "Learning single-image 3D reconstruction by generative modelling of shape, pose and shading," *Int. J. Comput. Vis.*, vol. 128, pp. 835–854, 2019.
- [34] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond Pascal: A benchmark for 3D object detection in the wild," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2014, pp. 75–82.
- [35] O. Faugeras, Q.-T. Luong, and T. Papadopoulos, *The Geometry of Multiple Images*. Cambridge, MA, USA: MIT Press, 2001.
- [36] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6612–6619.
- [37] B. Ummenhofer *et al.*, "Demon: Depth and motion network for learning monocular stereo," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5622–5631.
- [38] C. Godard, O. M. Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 270–279.
- [39] C. Bregler, A. Hertzmann, and H. Biermann, "Recovering non-rigid 3D shape from image streams," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2000, pp. 690–696.
- [40] D. Novotny, N. Ravi, B. Graham, N. Neverova, and A. Vedaldi, "C3DPO: Canonical 3D pose networks for non-rigid structure from motion," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7687–7696.
- [41] B. K. P. Horn and M. J. Brooks, *Shape from Shading*. Cambridge, MA, USA: MIT Press, 1989.
- [42] R. Zhang, P.-S. Tsai, J. E. Cryer, and M. Shah, "Shape-from-shading: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 8, pp. 690–706, Aug. 1999.
- [43] J. J. Koenderink, "What does the occluding contour tell us about solid shape?," *Perception*, vol. 13, no. 3, pp. 321–330, 1984.
- [44] A. P. Witkin, "Recovering surface shape and orientation from texture," *Artif. Intell.*, vol. 17, no. 1–3, pp. 17–45, 1981.
- [45] P. Agrawal, J. Carreira, and J. Malik, "Learning to see by moving," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 37–45.
- [46] D. Novotny, D. Larlus, and A. Vedaldi, "Learning 3D object categories by looking around them," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5228–5237.
- [47] C. Wang, J. M. Buenaposada, R. Zhu, and S. Lucey, "Learning depth from monocular videos using direct methods," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2022–2030.
- [48] X. Li *et al.*, "Online adaptation for consistent mesh reconstruction in the wild," in *Proc. Neural Inf. Process. Syst.*, 2020, pp. 15009–15019. [Online]. Available: <https://papers.nips.cc/paper/2020/hash/aba3b6fd5d186d28e06ff97135cade7f-Abstract.html>
- [49] Y. Luo *et al.*, "Single view stereo matching," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 155–163.
- [50] V. Sitzmann, M. Zollhöfer, and G. Wetzstein, "Scene representation networks: Continuous 3D-structure-aware neural scene representations," in *Proc. Neural Inf. Process. Syst.*, 2019, pp. 1119–1130.
- [51] O. Wiles, G. Gkioxari, R. Szeliski, and J. Johnson, "SynSin: End-to-end view synthesis from a single image," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 7467–7477.
- [52] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing scenes as neural radiance fields for view synthesis," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 405–421. [Online]. Available: [https://link.springer.com/chapter/10.1007/978-3-030-58452-8\\_24](https://link.springer.com/chapter/10.1007/978-3-030-58452-8_24)
- [53] M. Niemeyer, L. Mescheder, M. Oechsle, and A. Geiger, "Differentiable volumetric rendering: Learning implicit 3D representations without 3D supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 3504–3515.
- [54] L. Yariv *et al.*, "Multiview neural surface reconstruction by disentangling geometry and appearance," in *Proc. Neural Inf. Process. Syst.*, 2020, pp. 2492–2502.
- [55] S. Suwajanakorn, N. Snavely, J. Tompson, and M. Norouzi, "Discovery of latent 3D keypoints via end-to-end geometric reasoning," in *Proc. Neural Inf. Process. Syst.*, 2018, pp. 2063–2074. [Online]. Available: <https://dblp.org/rec/conf/nips/SuwajanakornSTN18.html?view=bibtex>
- [56] C.-H. Chen *et al.*, "Unsupervised 3D pose estimation with geometric self-supervision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5714–5724.
- [57] A. T. Tran, T. Hassner, I. Masi, and G. Medioni, "Regressing robust and discriminative 3D morphable models with a very deep neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5163–5172.
- [58] A. T. Tran, T. Hassner, I. Masi, E. Paz, Y. Nirkin, and G. Medioni, "Extreme 3D face reconstruction: Seeing through occlusions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3935–3944.
- [59] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou, "Joint 3D face reconstruction and dense alignment with position map regression network," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 557–574.
- [60] J. Guo, X. Zhu, Y. Yang, F. Yang, Z. Lei, and S. Z. Li, "Towards fast, accurate and stable 3D dense face alignment," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 152–168.
- [61] Y. Feng, H. Feng, M. J. Black, and T. Bolkart, "Learning an animatable detailed 3D face model from in-the-wild images," 2020, *arXiv:2012.04012*.
- [62] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "SMPL: A skinned multi-person linear model," *ACM Trans. Graph.*, vol. 34, no. 6, 2015, Art. no. 248.

- [63] J. Thewlis, H. Bilen, and A. Vedaldi, "Unsupervised learning of object frames by dense equivariant image labelling," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 844–855. [Online]. Available: <https://dblp.org/rec/conf/nips/ThewlisBV17.html?view=bibtex>
- [64] J. Thewlis, H. Bilen, and A. Vedaldi, "Modelling and unsupervised learning of symmetric deformable object categories," in *Proc. Neural Inf. Process. Syst.*, 2018, pp. 8189–8200.
- [65] Z. Shu, M. Sahasrabudhe, A. Guler, D. Samaras, N. Paragios, and I. Kokkinos, "Deforming autoencoders: Unsupervised disentangling of shape and appearance," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 650–665.
- [66] J.-Y. Zhu *et al.*, "Visual object networks: Image generation with disentangled 3D representations," in *Proc. Neural Inf. Process. Syst.*, 2018, pp. 118–129. [Online]. Available: <https://dblp.org/rec/conf/nips/ZhuZZ00TF18.html?view=bibtex>
- [67] M. M. Loper and M. J. Black, "OpenDR: An approximate differentiable renderer," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 154–169.
- [68] H. Kato, Y. Ushiku, and T. Harada, "Neural 3D mesh renderer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3907–3916.
- [69] S. Liu, T. Li, W. Chen, and H. Li, "Soft rasterizer: A differentiable renderer for image-based 3D reasoning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7707–7716.
- [70] B. Horn, "Obtaining shape from shading information," in *The Psychology of Computer Vision*. New York, NY, USA: McGraw-Hill, 1975.
- [71] P. N. Belhumeur, D. J. Kriegman, and A. L. Yuille, "The bas-relief ambiguity," *Int. J. Comput. Vis.*, vol. 35, pp. 33–44, 1999.
- [72] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 5574–5584. [Online]. Available: <https://dblp.org/rec/conf/nips/KendallG17.html?view=bibtex>
- [73] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015. [Online]. Available: <https://dblp.org/rec/journals/corr/SimonyanZ14a.html?view=bibtex>
- [74] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. Neural Inf. Process. Syst.*, 2015, pp. 2017–2025.
- [75] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3730–3738.
- [76] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-PIE," *Image Vis. Comput.*, vol. 28, no. 5, pp. 807–813, 2010.
- [77] L. A. Jeni, J. F. Cohn, and T. Kanade, "Dense 3D face alignment from 2D videos in real-time," in *Proc. Int. Conf. Autom. Face Gesture Recognit.*, 2015, pp. 1–8.
- [78] X. Zhang *et al.*, "BP4D-Spontaneous: A high-resolution spontaneous 3D dynamic facial expression database," *Image Vis. Comput.*, vol. 32, no. 10, pp. 692–706, 2014.
- [79] L. Yin, X. Chen, Y. Sun, T. Worm, and M. Reale, "A high-resolution 3D dynamic facial expression database," in *Proc. Int. Conf. Autom. Face Gesture Recognit.*, 2008, pp. 1–6.
- [80] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.
- [81] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "SUN database: Large-scale scene recognition from abbey to zoo," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 3485–3492.
- [82] W. Zhang, J. Sun, and X. Tang, "Cat head detection - How to effectively exploit shape and texture features," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 802–816.
- [83] O. M. Parkhi, A. Vedaldi, A. Zisserman, and C. V. Jawahar, "Cats and dogs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3498–3505.
- [84] A. X. Chang *et al.*, "Shapenet: An information-rich 3D model repository," 2015, *arXiv:1512.03012*.
- [85] D. Eigen, C. Puhrsch, and R. Fergus, "Depth map prediction from a single image using a multi-scale deep network," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 2366–2374.
- [86] E. J. Crowley, O. M. Parkhi, and A. Zisserman, "Face painting: Querying art with photos," in *Proc. Brit. Mach. Vis. Conf.*, 2015, pp. 1–13.
- [87] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep speaker recognition," in *Proc. INTERSPEECH*, 2018, pp. 1086–1090.
- [88] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Representations*, 2018. [Online]. Available: <https://dblp.org/rec/conf/iclr/GidarisSK18.html?view=bibtex>
- [89] H.-Y. F. Tung, A. W. Harley, W. Seto, and K. Fragkiadaki, "Adversarial inverse graphics networks: Learning 2D-to-3D lifting and image-to-image translation from unpaired supervision," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 4364–4372.
- [90] A. Tewari *et al.*, "MoFA: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3735–3744.
- [91] Z.-H. Feng *et al.*, "Evaluation of dense 3D reconstruction from 2D face images in the wild," in *Proc. Int. Conf. Autom. Face Gesture Recognit.*, 2018, pp. 780–786.
- [92] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results," *Image Vis. Comput.*, vol. 47, pp. 3–18, 2016.



**Shangzhe Wu** received the bachelor's degree in computer science from the Hong Kong University of Science and Technology, where he worked with Chi-Keung Tang and Yu-Wing Tai on image translation. He is currently working toward the DPhil degree with the Visual Geometry Group, University of Oxford, supervised by Andrea Vedaldi. His research focuses on unsupervised 3D understanding. He was the recipient of the Best Paper Award at CVPR 2020.



**Christian Rupprecht** received the PhD degree from the Technical University of Munich, Germany, advised by Nassir Navab and Gregory D. Hager (JHU). He is currently a postdoctoral researcher with the Visual Geometry Group, University of Oxford. For six months, he was with Chris Pal, Mila Institute, Montreal, working on AI safety. His research interests include self-supervised and minimally supervised learning for computer vision.



**Andrea Vedaldi** is currently a professor of computer vision and machine learning with the University of Oxford, where he has been co-leading Visual Geometry Group since 2012. He is also a research scientist with Facebook AI Research, London. He has authored or coauthored more than 130 peer-reviewed publications in the top machine vision and artificial intelligence conferences and journals. His research interests include unsupervised learning of representations and geometry in computer vision. He was the recipient of the Mark Everingham Prize for selfless contributions to the computer vision community, the Open Source Software Award by the ACM, and the Best Paper Award from the Conference on Computer Vision and Pattern Recognition.

► **For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/csdl](http://www.computer.org/csdl).**