Fake it till you make it: face analysis in the wild using synthetic data alone

Erroll Wood* Tadas Baltrušaitis* Charlie Hewitt
Sebastian Dziadzio Thomas J. Cashman Jamie Shotton
Microsoft

Abstract

We demonstrate that it is possible to perform face-related computer vision in the wild using synthetic data alone. The community has long enjoyed the benefits of synthesizing training data with graphics, but the domain gap between real and synthetic data has remained a problem, especially for human faces. Researchers have tried to bridge this gap with data mixing, domain adaptation, and domain-adversarial training, but we show that it is possible to synthesize data with minimal domain gap, so that models trained on synthetic data generalize to real in-the-wild datasets. We describe how to combine a procedurally-generated parametric 3D face model with a comprehensive library of hand-crafted assets to render training images with unprecedented realism and diversity. We train machine learning systems for face-related tasks such as landmark localization and face parsing, showing that synthetic data can both match real data in accuracy as well as open up new approaches where manual labeling would be impossible.

1. Introduction

When faced with a machine learning problem, the hardest challenge often isn't choosing the right machine learning model, it's finding the right data. This is especially difficult in the realm of human-related computer vision, where concerns about the fairness of models and the ethics of deployment are paramount [31]. Instead of collecting and labelling real data, which is slow, expensive, and subject to bias, it can be preferable to *synthesize* training data using computer graphics [68]. With synthetic data, you can guarantee perfect labels without annotation noise, generate rich labels that are otherwise impossible to label by hand, and have full control over variation and diversity in a dataset.

Rendering convincing humans is one of the hardest problems in computer graphics. Movies and video games have shown that realistic digital humans are possible, but with significant artist effort per individual [22, 26]. While it's possible to generate endless novel face images with recent self-supervised approaches [27], corresponding labels for supervised learning are not available. As a result, previous work has resorted to synthesizing facial training data with simplifications, with results that are far from realistic. We have seen progress in efforts that attempt to cross the domain gap using domain adaptation [60] by refining synthetic images to look more real, and domain-adversarial training [13] where machine learning models are encouraged to ignore differences between the synthetic and real domains, but less work has attempted to improve the quality of synthetic data itself. Synthesizing realistic face data has been considered so hard that we encounter the assumption that synthetic data cannot fully replace real data for problems in the wild [60].

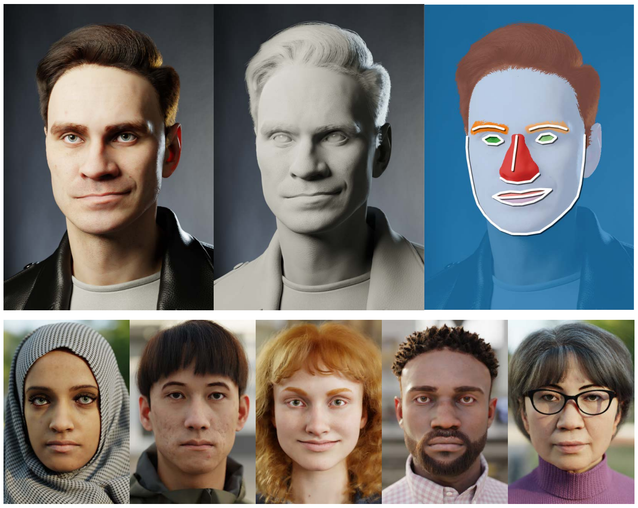

Figure 1. We render training images of faces with unprecedented realism and diversity. The first example above is shown along with 3D geometry and accompanying labels for machine learning.

In this paper we demonstrate that the opportunities for synthetic data are much wider than previously realised, and are achievable today. We present a new method of acquiring training data for faces – rendering 3D face models with an unprecedented level of realism and diversity (see Figure 1). With a sufficiently good synthetic framework, it is possible

*Denotes equal contribution.
https://microsoft.github.io/FaceSynthetics

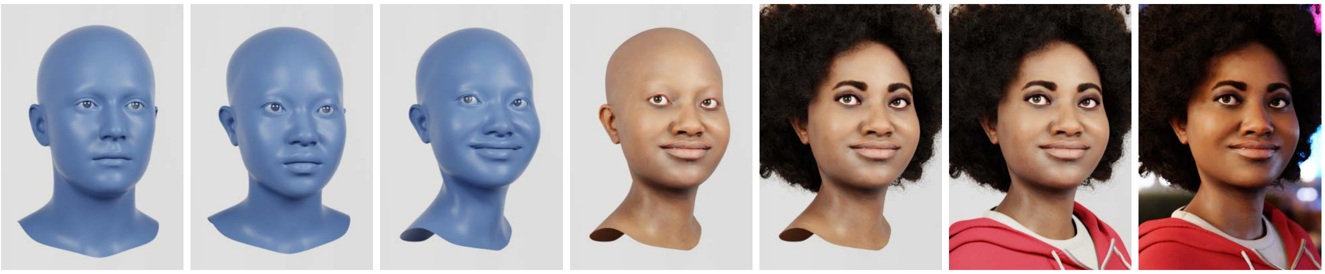

Figure 2. We procedurally construct synthetic faces that are realistic and expressive. Starting with our template face, we randomize the identity, choose a random expression, apply a random texture, attach random hair and clothing, and render the face in a random environment.

to create training data that can be used to solve real world problems in the wild, without using any real data at all.

It requires considerable expertise and investment to develop a synthetics framework with minimal domain gap. However, once implemented, it becomes possible to generate a wide variety of training data with minimal incremental effort. Let's consider some examples; say you have spent time labelling face images with landmarks. However, you suddenly require additional landmarks in each image. Relabelling and verifying will take a long time, but with synthetics, you can regenerate clean and consistent labels at a moment's notice. Or, say you are developing computer vision algorithms for a new camera, e.g. an infrared face-recognition camera in a mobile phone. Few, if any, hardware prototypes may exist, making it hard to collect a dataset. Synthetics lets you render faces from a simulated device to develop algorithms and even guide hardware design itself.

We synthesize face images by procedurally combining a parametric face model with a large library of high-quality artist-created assets, including textures, hair, and clothing (see Figure 2). With this data we train models for common face-related tasks: face parsing and landmark localization. Our experiments show that models trained with a single generic synthetic dataset can be just as accurate as those trained with task-specific real datasets, achieving results in line with the state of the art. This opens the door to other face-related tasks that can be confidently addressed with synthetic data instead of real.

Our contributions are as follows. First, we describe how to synthesize realistic and diverse training data for face analysis in the wild, achieving results in line with the state of the art. Second, we present ablation studies that validate the steps taken to achieve photorealism. Third is the synthetic dataset itself, which is available from our project webpage: https://microsoft.github.io/FaceSynthetics.

2. Related work

Diverse face datasets are very difficult to collect and annotate. Collection techniques such as web crawling pose significant privacy and copyright concerns. Manual annotation is error-prone and can often result in inconsistent labels. Hence, the research community is increasingly looking at augmenting or replacing real data with synthetic.

2.1. Synthetic face data

The computer vision community has used synthetic data for many tasks, including object recognition [23, 44, 51, 73], scene understanding [12, 25, 47, 50], eye tracking [63, 68], hand tracking [40, 61], and full-body analysis [41, 59, 65]. However, relatively little previous work has attempted to generate full-face synthetics using computer graphics, due to the complexity of modeling the human head.

A common approach is to use a 3D Morphable Model (3DMM) [5], since these can provide consistent labels for different faces. Previous work has focused on parts of the face such as the eye region [62] or the *hockey mask* [45, 76]. Zeng et al. [76], Richardson et al. [46], and Sela et al. [58] used 3DMMs to render training data for reconstructing detailed facial geometry. Similarly, Wood et al. [69] rendered an eye region 3DMM for gaze estimation. However, since these approaches only render part of the face, the resulting data has limited use for tasks that consider the whole face.

Building parametric models is challenging, so an alternative is to render 3D scans directly [4, 55, 62, 68]. Jeni et al. [24] rendered the BU-4DFE dataset [74] for dense 3D face alignment, and Kuhnke and Ostermann [30] rendered commercially-available 3D head scans for head pose estimation. While often realistic, these approaches are limited by the diversity expressed in the scans themselves, and cannot provide rich semantic labels for machine learning.

Manipulating 2D images can be an alternative to using a 3D graphics pipeline. Zhu et al. [79] fit a 3DMM to face images, and warped them to augment the head pose. Nojavanasghari et al. [42] composited hand images onto faces to improve face detection. These approaches can only make minor adjustments to existing images, limiting their use.

2.2. Training with synthetic data

Although it is common to rely on synthetic data alone for full-body tasks [54, 59], synthetic data is rarely used on its own for face-related machine learning. Instead it is either first adapted to make it look more like some target domain, or used alongside real data for pre-training [76] or regularizing models [16, 29]. The reason for this is the *domain gap* – a difference in distributions between real and synthetic data which makes generalization difficult [25].

Learned domain adaptation modifies synthetic images to better match the appearance of real images. Shrivastava et al. [60] use an adversarial refiner network to adapt synthetic eye images with regularization to preserve annotations. Similarly, Bak et al. [3] adapt synthetic data using a CycleGAN [77] with a regularization term for preserving identities. A limitation of learned domain adaptation is the tendency for image semantics to change during adaptation [15], hence the need for regularization [3, 40, 60]. These techniques are therefore unsuitable for fine-grained annotations, such as per-pixel labels or precise landmark coordinates.

Instead of adapting data, it is possible to learn features that are resistant to the differences between domains [13, 57]. Wu et al. [71] mix real and synthetic data through a domain classifier to learn domain-invariant features for text detection, and Saleh et al. [56] exploit the observation that shape is less affected by the domain gap than appearance for scene semantic segmentation.

In our work, we do not perform any of these techniques and instead minimize the domain gap at the source, by generating highly realistic synthetic data.

3. Synthesizing face images

The Visual Effects (VFX) industry has developed many techniques for convincing audiences that 3D faces are real, and we build upon these in our approach. However, a key difference is scale: while VFX might be used for a handful of actors, we require diverse training data of thousands of synthetic individuals. To address this, we use procedural generation to randomly create and render novel 3D faces without any manual intervention.

We start by sampling a generative 3D face model that captures the diversity of the human population. We then randomly ‘dress up’ each face with samples from large collections of hair, clothing, and accessory assets. All collections are sampled independently to create synthetic individuals who are as diverse as possible from one another. This section describes the technical components we built in order to enable asset collections that can be mixed-and-matched atop 3D faces in a random, yet plausible manner.

3.1. 3D face model

Our generative 3D face model captures how face shape varies across the human population, and changes during facial expressions. It is a blendshape-based face rig similar to previous work [17, 34], and comprises a mesh of $N = 7,667$ vertices and $7,414$ polygons, and a minimal skeleton of $K = 4$ joints: the head, neck, and two eyes.

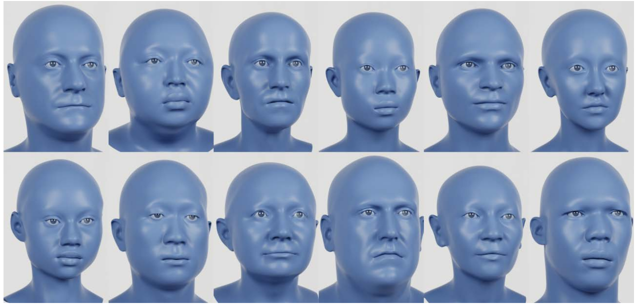

Figure 3. 3D faces sampled from our generative model, demonstrating how our model captures the diversity of the human population.

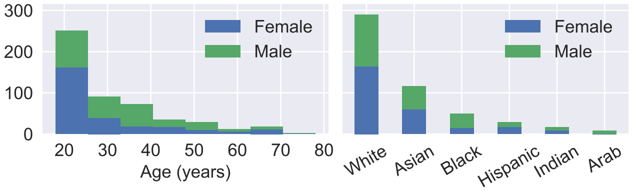

Figure 4. Histograms of self-reported age and ethnicity in our scan collection, which was used to build our face model and texture library. Our collection covers a range of age and ethnicity.

The face mesh vertex positions are defined by mesh generating function $\mathcal{M}(\vec{\beta}, \vec{\psi}, \vec{\theta}) : \mathbb{R}^{|\vec{\beta}| \times |\vec{\psi}| \times |\vec{\theta}|} \rightarrow \mathbb{R}^{N \times 3}$ which takes parameters $\vec{\beta} \in \mathbb{R}^{|\vec{\beta}|}$ for identity, $\vec{\psi} \in \mathbb{R}^{|\vec{\psi}|}$ for expression, and $\vec{\theta} \in \mathbb{R}^{K \times 3}$ for skeletal pose. The pose parameters $\vec{\theta}$ are per-joint local rotations represented as Euler angles. $\mathcal{M}$ is defined as

$$\mathcal{M}(\vec{\beta}, \vec{\psi}, \vec{\theta}) = \mathcal{L}(\mathcal{T}(\vec{\beta}, \vec{\psi}), \vec{\theta}, \mathcal{J}(\vec{\beta}); \mathbf{W})$$

where $\mathcal{L}(\mathbf{X}, \vec{\theta}, \mathbf{J}; \mathbf{W})$ is a standard linear blend skinning (LBS) function [33] that rotates vertex positions $\mathbf{X} \in \mathbb{R}^{N \times 3}$ about joint locations $\mathbf{J} \in \mathbb{R}^{K \times 3}$ by local joint rotations $\vec{\theta}$, with per-vertex weights $\mathbf{W} \in \mathbb{R}^{K \times N}$ determining how rotations are interpolated across the mesh. $\mathcal{T}(\vec{\beta}, \vec{\psi}) : \mathbb{R}^{|\vec{\beta}| \times |\vec{\psi}|} \rightarrow \mathbb{R}^{N \times 3}$ constructs a face mesh in the bind pose by adding displacements to the template mesh $\overline{\mathbf{T}} \in \mathbb{R}^{N \times 3}$, which represents the average face with neutral expression:

$$\mathcal{T}(\vec{\beta}, \vec{\psi})^j_k = \overline{T}^j_k + \beta_i S^{ij}_k + \psi_i E^{ij}_k$$

given linear identity basis $\mathbf{S} \in \mathbb{R}^{|\vec{\beta}| \times N \times 3}$ and expression basis $\mathbf{E} \in \mathbb{R}^{|\vec{\psi}| \times N \times 3}$. Note the use of Einstein summation notation in this definition and below. Finally, $\mathcal{J}(\vec{\beta}) : \mathbb{R}^{|\vec{\beta}|} \rightarrow \mathbb{R}^{K \times 3}$ moves the template joint locations $\overline{\mathbf{J}} \in \mathbb{R}^{K \times 3}$ to account for changes in identity:

$$\mathcal{J}(\vec{\beta})^j_k = \overline{J}^j_k + W^j_l \beta_i S^{il}_k.$$

We learn the identity basis $\mathbf{S}$ from high quality 3D scans of $M = 511$ individuals with neutral expression. Each scan

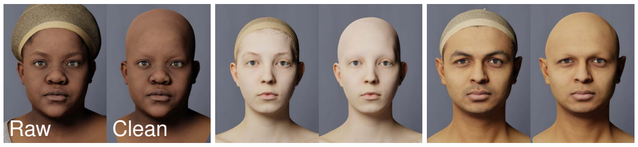

Figure 5. We manually "clean" raw high-resolution 3D head scans to remove noise and hair. We use the resulting clean scans to build our generative geometry model and texture library.

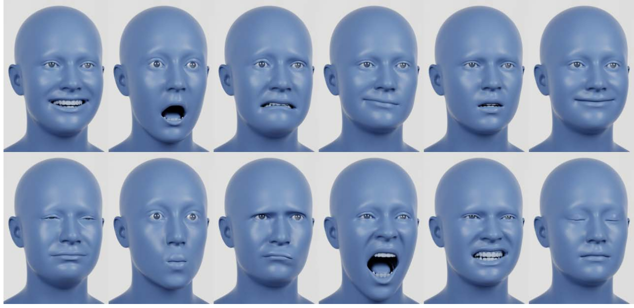

Figure 6. Examples from our data-driven expression library and manually animated sequence, visualized on our template face.

was cleaned (see Figure 5), and registered to the topology of $\overline{\mathbf{T}}$ using commercial software [52], resulting in training dataset $\mathbf{V} \in \mathbb{R}^{M \times 3N}$. We then jointly fit identity basis $\mathbf{S}$ and parameters $[\vec{\beta}_1, \ldots, \vec{\beta}_M]$ to $\mathbf{V}$. In order to generate novel face shapes, we fit a multivariate normal distribution to the fitted identity parameters, and sample from it (see Figure 3). As is common in computer animation, both expression basis $\mathbf{E}$ and skinning weights $\mathbf{W}$ were authored by an artist, and are kept fixed while learning $\mathbf{S}$.

3.2. Expression

We apply random expressions to each face so that our downstream machine learning models are robust to facial motion. We use two sources of facial expression. Our primary source is a library of 27,000 expression parameters $\{\vec{\psi}_i\}$ built by fitting a 3D face model to a corpus of 2D images with annotated face landmarks. However, since the annotated landmarks are sparse, it is not possible to recover all types of expression from these landmarks alone, e.g. cheek puffs. Therefore, we additionally sample expressions from a manually animated sequence that was designed to fill the gaps in our expression library by exercising the face in realistic, but extreme ways. Figure 6 shows samples from our expression collection. In addition to facial expression, we layer random eye gaze directions on top of sampled expressions, and use procedural logic to pose the eyelids accordingly.

3.3. Texture

Synthetic faces should look realistic even when viewed at extremely close range, for example by an eye-tracking camera in a head-mounted device. To achieve this, we collected 200 sets of high resolution (8192×8192 px) textures

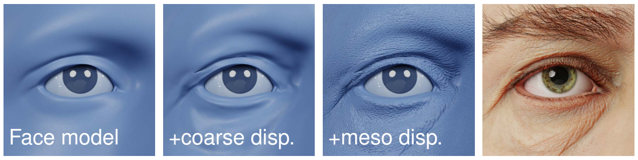

Figure 7. We apply coarse and meso-displacement to our 3D face model to ensure faces look realistic even when viewed close-up.

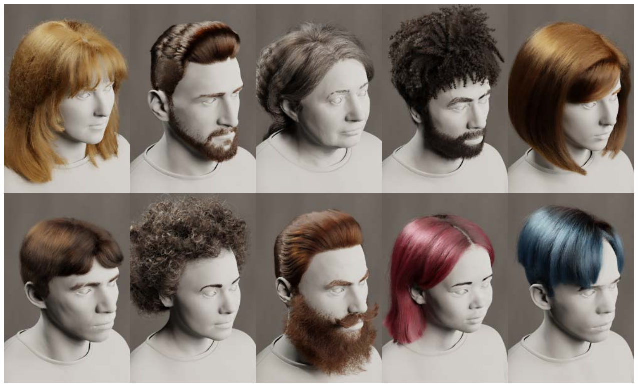

Figure 8. Our hair library contains a diverse range of scalp hair, eyebrows, and beards. When assembling a 3D face, we choose hair style and appearance at random.

from our cleaned face scans. For each scan, we extract one albedo texture for skin color, and two displacement maps (see Figure 7). The coarse displacement map encodes scan geometry that is not captured by the sparse nature of our vertex-level identity model. The meso-displacement map approximates skin-pore level detail and is built by high-pass filtering the albedo texture, assuming that dark pixels correspond to slightly recessed parts of the skin.

Unlike previous work [45, 76], we do not build a generative model of texture, as such models struggle to faithfully produce high-frequency details like wrinkles and pores. Instead, we simply pick a corresponding set of albedo and displacement textures from each scan. The textures are combined in a physically-based skin material featuring subsurface scattering [9]. Finally, we optionally apply makeup effects to simulate eyeshadow, eyeliner and mascara.

3.4. Hair

In contrast to other work which approximates hair with textures or coarse geometry [17, 55], we represent hair as individual 3D strands, with a full head of hair comprising over 100,000 strands. Modelling hair at the strand level allows us to capture realistic multi-path illumination effects. Shown in Figure 8, our hair library includes 512 scalp hair styles, 162 eyebrows, 142 beards, and 42 sets of eyelashes. Each asset was authored by a groom artist who specializes in creating digital hair. At render time, we randomly combine scalp, eyebrow, beard, and eyelash grooms.

We use a physically-based procedural hair shader to ac-

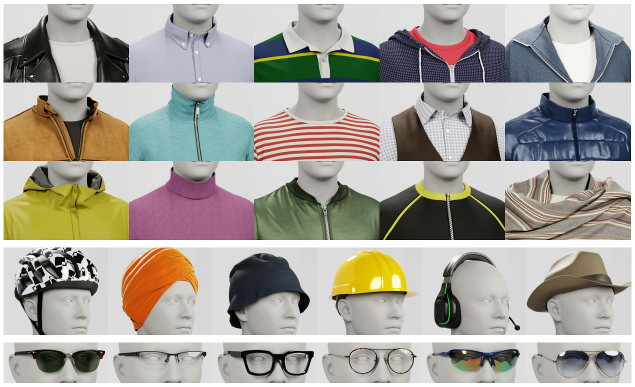

Figure 9. Each face is dressed in a random outfit assembled from our digital wardrobe – a collection of diverse 3D clothing and accessory assets that can be fit around our 3D head model.

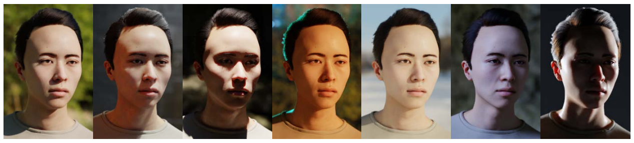

Figure 10. We use HDRIs to illuminate the face. The same face can look very different under different illumination.

curately model the complex material properties of hair [8]. This shader allows us to control the color of the hair with parameters for melanin [38] and grayness, and even lets us dye or bleach the hair for less common hair styles.

3.5. Clothing

Images of faces often include what someone is wearing, so we dress our faces in 3D clothing. Our digital wardrobe contains 30 upper-body outfits which were manually created using clothing design and simulation software [10]. As shown in Figure 9, these outfits include formal, casual, and athletic clothing. In addition to upper-body garments, we dress our faces in headwear (36 items), facewear (7 items) and eyewear (11 items) including helmets, head scarves, face masks, and eyeglasses. All clothing items were authored on an unclothed body mesh with either the average male or female body proportions [37] in a relaxed stance.

We deform garments with a non-rigid cage-based deformation technique [2] so they fit snugly around different shaped faces. Eyeglasses are rigged with a skeleton, and posed using inverse kinematics so the temples and nose-bridge rest on the corresponding parts of the face.

3.6. Rendering

We render face images with Cycles, a photorealistic ray-tracing renderer [6]. We randomly position a camera around the head, and point it towards the face. The focal length and depth of field are varied to simulate different cameras and lenses. We employ image-based lighting [11] with high dynamic range images (HDRI) to illuminate the face and provide a background (see Figure 10). For each image, we randomly pick from a collection of 448 HDRIs that include a range of different environments [75]. See Figure 11 for examples of faces rendered with our framework.

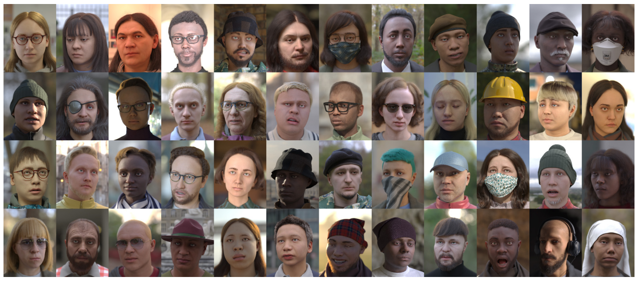

Figure 11. Examples of synthetic faces that we randomly generated and rendered for use as training data.

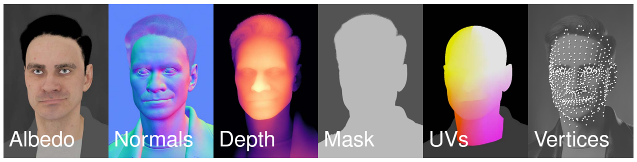

Figure 12. We also synthesize labels for machine learning. Above are additional label types beyond those shown in Figure 1.

In addition to rendering color images, we generate ground truth labels (see Figure 12). While our experiments in section 4 focus on landmark and segmentation annotations, synthetics lets us easily create a variety of rich and accurate labels that enable new face-related tasks (see subsection 4.5).

4. Face analysis

We evaluate our synthetic data on two common face analysis tasks: face parsing and landmark localization. We show that models trained on our synthetic data demonstrate competitive performance to the state of the art. Note that all evaluations using our models are *cross-dataset* – we train purely on synthetic data and test on real data, while the state of the art evaluates *within-dataset*, allowing the models to learn potential biases in the data.

4.1. Training methodology

We render a **single** training dataset for both landmark localization and face parsing, comprising 100,000 images at 512×512 resolution. It took 48 hours to render using 150 NVIDIA M60 GPUs.

During training, we perform data augmentation including rotations, perspective warps, blurs, modulations to brightness and contrast, addition of noise, and conversion to grayscale. Such augmentations are especially important for synthetic images which are otherwise free of imperfection (see subsection 4.4). While some of these could be done at render time, we perform them at training time in order to randomly

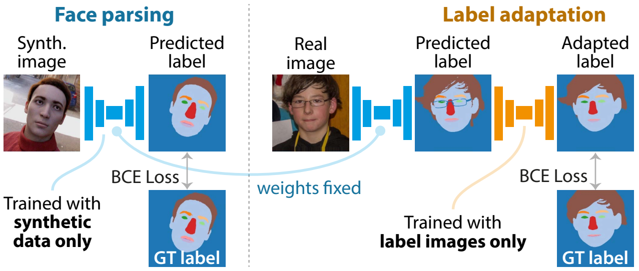

Figure 13. We train a face parsing network (using synthetic data only) followed by a label adaptation network to address systematic differences between synthetic and human-annotated labels.

apply different augmentations to the same training image. We implemented neural networks with PyTorch [43], and trained them with the Adam optimizer [28].

4.2. Face parsing

Face parsing assigns a class label to each pixel in an image, e.g. skin, eyes, mouth, or nose. We evaluate our synthetic training data on two face parsing datasets: **Helen** [32] is the best-known benchmark in the literature. It contains 2,000 training images, 230 validation images, and 100 testing images, each with 11 classes. Due to labelling errors in the original dataset, we use Helen* [35], a popular rectified version of the dataset which features corrected training labels, but leaves testing labels unmodified for a fair comparison. **LaPa** [36] is a recently-released dataset which uses the same labels as Helen, but has more images, and exhibits more challenging expressions, poses, and occlusions. It contains 18,176 training images, 2,000 validation images and 2,000 testing images.

As is common [35, 36], we use the provided 2D landmarks to align faces before processing. We scale and crop each image so the landmarks are centered in a 512 × 512px region of interest. Following prediction, we undo this transform to compute results against the original label annotation, without any resizing or cropping.

Method We treat face parsing as image-to-image translation. Given an input color image x containing C classes, we wish to predict a C-channel label image $\hat{y}$ of the same spatial dimensions that matches the ground truth label image y. Pixels in y are one-hot encoded with the index of the true class. For this, we use a UNet [49] with ResNet-18 encoder [21, 72]. We train this network with synthetic data only, minimizing a binary cross-entropy (BCE) loss between predicted and ground truth label images. Note that there is nothing novel about our choice of architecture or loss function, this is a well-understood approach for this task.

Label adaptation. There are bound to be minor systematic differences between synthetic labels and human-annotated labels. For example, where exactly is the boundary between the nose and the rest of the face? To evaluate our synthetic data without needing to carefully tweak our synthetic label generation process for a specific real dataset, we use *label adaptation*. Label adaptation transforms labels predicted by our face parsing network (trained with synthetic data alone) into labels that are closer to the distribution in the real dataset (see Figure 13). We treat label adaptation as another image-to-image translation task, and use a UNet with ResNet18 encoder [72]. To ensure this stage is not able to 'cheat', it is trained only on pairs of predicted labels $\hat{y}$ and ground truth labels y. It is trained entirely separately from the face parsing network, and never sees any real images.

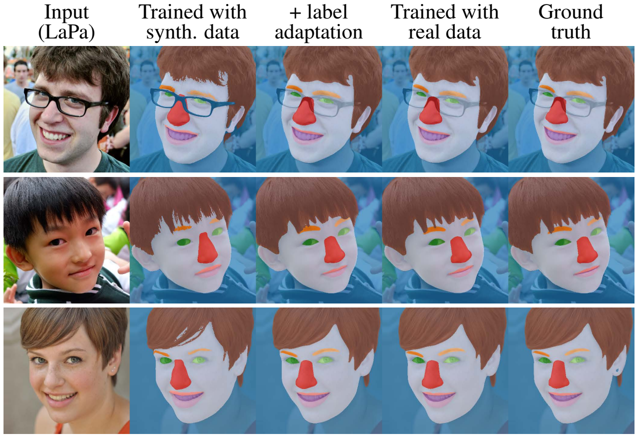

Figure 14. Face parsing results by networks trained with synthetic data (with and without label adaptation) and real data. Label adaptation addresses systematic differences between synthetic and real labels, e.g. the shape of the nose class, or granularity of hair.

Results See Tables 1 and 2 for comparisons against the state of the art, and Figure 14 for some example predictions. Although networks trained with our generic synthetic data do not outperform the state of the art, it is notable that they achieve similar results to previous work trained within-dataset on task-specific data.

Comparison to real data. We also trained a network on the training portion of each real dataset to separate our training methodology from our synthetic data, presented as "Ours (real)" in Tables 1 and 2. It can be seen that training with synthetic data alone produces comparable results to training with real data.

4.3. Landmark localization

Landmark localization finds the position of facial points of interest in 2D. We evaluate our approach on the **300W** [53] dataset, which is split into common (554 images), challenging (135 images) and private (600 images) subsets.

Method We train a ResNet34 [21] with mean squared error loss to directly predict 68 2D landmark coordinates per-image. We use the provided bounding boxes to extract a 256 × 256 pixel region-of-interest from each image. The private set has no bounding boxes, so we use a tight crop

Table 1. A comparison with the state of the art on the Helen dataset, using F_1 score. As is common, scores for hair and other fine-grained categories are omitted to aid comparison to previous work. The overall score is computed by merging the nose, brows, eyes, and mouth categories. Training with our synthetic data achieves results in line with the state of the art, trained with real data.

Method		Skin	Nose	Upper lip	Inner mouth	Lower lip	Brows	Eyes	Mouth	Overall
Guo et al. [19]	AAAI'18	93.8	94.1	75.8	83.7	83.1	80.4	87.1	92.4	90.5
Wei et al. [67]	TIP'19	95.6	95.2	80.0	86.7	86.4	82.6	89.0	93.6	91.6
Lin et al. [35]	CVPR'19	94.5	95.6	79.6	86.7	89.8	83.1	89.6	95.0	92.4
Liu et al. [36]	AAAI'20	94.9	95.8	83.7	89.1	91.4	83.5	89.8	96.1	93.1
Te et al. [64]	ECCV'20	94.6	96.1	83.6	89.8	91.0	90.2	84.9	95.5	93.2
Ours (real)		95.1	94.7	81.6	87.0	88.9	81.5	87.6	94.8	91.6
Ours (synthetic)		95.1	94.5	82.3	89.1	89.9	83.5	87.3	95.1	92.0

Table 2. A comparison with the state of the art on LaPa, using F_1 score. For eyes and brows, L and R are left and right. For lips, U, I, and L are upper, inner, and lower. Training with our synthetic data achieves results in line with the state of the art, trained with real data.

Method		Skin	Hair	L-eye	R-eye	U-lip	I-mouth	L-lip	Nose	L-Brow	R-Brow	Mean
Liu et al. [36]	AAAI'20	97.2	96.3	88.1	88.0	84.4	87.6	85.7	95.5	87.7	87.6	89.8
Te et al. [64]	ECCV'20	97.3	96.2	89.5	90.0	88.1	90.0	89.0	97.1	86.5	87.0	91.1
Ours (real)		97.5	86.9	91.4	91.5	87.3	89.8	89.4	96.9	89.3	89.3	90.9
Ours (synthetic)		97.1	85.7	90.6	90.1	85.9	88.8	88.4	96.7	88.6	88.5	90.1

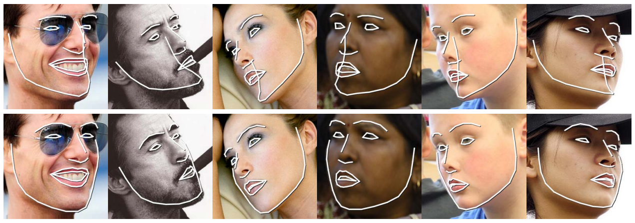

Figure 15. Predictions before (top) and after (bottom) label adaptation. The main difference is changing the jawline from a 3D-to-2D projection to instead follow the facial outline in the image.

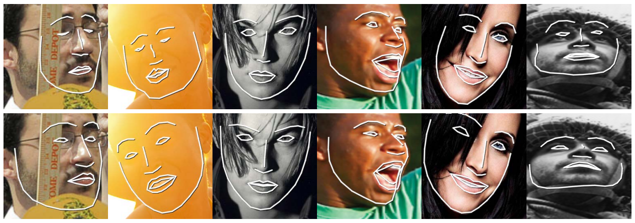

Figure 16. Predictions by networks trained with real (top) and synthetic data (bottom). Note how the synthetic data network generalizes better across expression, illumination, pose, and occlusion.

Table 3. Landmark localization results on the common, challenging, and private subsets of 300W. Lower is better in all cases. Note that 0.5 FR rate translates to 3 images, while 0.17 corresponds to 1.

		Common	Challenging	Private
Method		NME	NME	$FR_{10\%}$
DenseReg [20]	CVPR'17	-	-	3.67
LAB [70]	CVPR'18	2.98	5.19	0.83
AWING [66]	ICCV'19	2.72	4.52	0.33
ODN [78]	CVPR'19	3.56	6.67	-
LaplaceKL [48]	ICCV'19	3.19	6.87	-
3FabRec [7]	CVPR'20	3.36	5.74	0.17
Ours (real)		3.37	5.77	1.17
Ours (synthetic)		3.09	4.86	0.50
Ablation studies				
No augmentation		4.25	7.87	4.00
Appearance augmentation		3.93	6.80	1.83
No hair or clothing		3.36	5.37	2.17
No clothing		3.20	5.09	1.00
No label adaptation (synth.)		5.61	8.43	4.67
No label adaptation (real)		3.44	5.71	1.17

around landmarks.

Label adaptation is performed using a two-layer perceptron to address systematic differences between synthetic and real landmark labels (Figure 15). This network is never exposed to any real images during training.

Results As evaluation metrics we use: Normalized Mean Error (NME) [53] – normalized by inter-ocular outer eye distance; and Failure Rate below a 10% error threshold ($FR_{10\%}$). See Table 3 for comparisons against state of the art on 300W dataset. It is clear that the network trained with our synthetic data can detect landmarks with accuracy comparable to recent methods trained with real data.

Comparison to real data We apply our training methodology (including data augmentations and label adaptation) to the the training and validation portions of the 300W dataset, to more directly compare real and synthetic data. Table 3 clearly shows that training with synthetic data leads to better results, even when comparing to a model trained on real data and evaluated within-dataset.

4.4. Ablation studies

We investigate the effect of synthetic **dataset size** on landmark accuracy. Figure 17 shows that landmark localization improves as we increase the number of training images, before starting to plateau at 100,000 images.

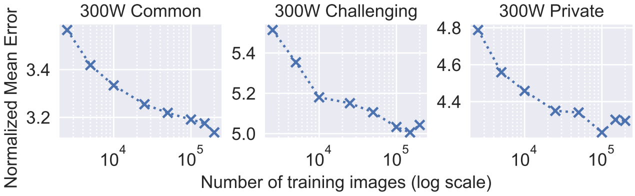

Figure 17. Landmark localization accuracy improves as we use more and more synthetic training data.

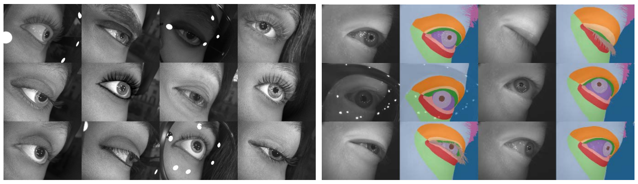

Figure 18. It is easy to generate synthetic training data for eye tracking (left) which generalizes well to real-world images (right).

We study the importance of **data augmentation** when training models on synthetic data. We train models with: 1) no augmentation; 2) appearance augmentation only (e.g. colour shifts, brightness and contrast); 3) full augmentation, varying both appearance and geometry (e.g. rotation and warping). Table 3 shows the importance of augmentation, without which synthetic data does not outperform real.

Table 3 also shows the importance of **label adaptation** when evaluating models trained on synthetic data – using label adaptation to improve label consistency reduces error. Adding label adaptation to a model trained on real data results in little change in performance, showing that it does not benefit already-consistent within-dataset labels.

If we remove **clothing and hair**, landmark accuracy suffers (Table 3). This verifies the importance of our hair library and digital wardrobe, which improve the realism of our data.

Additional ablation studies analyzing the impact of render quality, and variation in pose, expression, and identity can be found in the supplementary material.

4.5. Other examples

In addition to the quantitative results above, this section qualitatively demonstrates how we can solve additional problems using our synthetic face framework.

Eye tracking can be a key feature for virtual or augmented reality devices, but real training data can be difficult to acquire [14]. Since our faces look realistic close-up, it is easy for us to set up a synthetic eye tracking camera and render diverse training images, along with ground truth. Figure 18 shows example synthetic training data for such a camera, along with results for semantic segmentation.

Dense landmarks. In subsection 4.3, we presented results for localizing 68 facial landmarks. What if we wanted to predict ten times as many landmarks? It would be impossible for a human to annotate this many landmarks consistently and correctly. However, our approach lets us easily generate accurate dense landmark labels. Figure 19 shows the results of modifying our landmark network to regress 679 coordinates instead of 68, and training it with synthetic data.

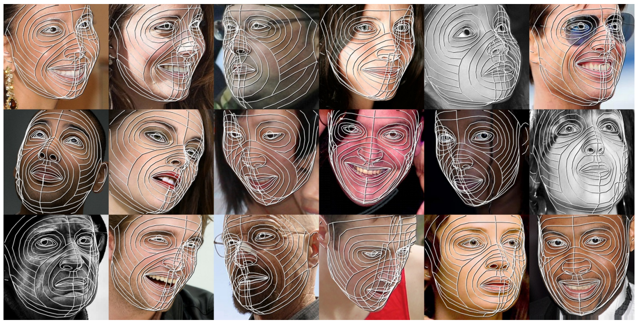

Figure 19. With synthetic data, we can easily train models that accurately predict ten times as many landmarks as usual. Here are some example dense landmark predictions on the 300W dataset.

4.6. Discussion

We have shown that it is possible to achieve results comparable with the state of the art for two well-trodden tasks: face parsing and landmark localization, without using a single real image during training. This is important since it opens the door to many other face-related tasks that can be addressed using synthetic data in the place of real data.

Limitations remain. As our parametric face model includes the head and neck only, we cannot simulate clothing with low necklines. We do not include expression-dependent wrinkling effects, so realism suffers during certain expressions. Since we sample parts of our model independently, we sometimes get unusual (but not impossible) combinations, such as feminine faces that have a beard. We plan to address these limitations with future work.

Photorealistic rendering is computationally expensive, so we must consider the environmental cost. In order to generate the dataset used in this paper, our GPU cluster used approximately 3,000kWh of electricity, equivalent to roughly 1.37 metric tonnes of CO_2, 100% of which was offset by our cloud computing provider. This impact is mitigated by the ongoing progress of cloud computing providers to become carbon negative and use renewable energy sources [1, 18, 39]. There is also the financial cost to consider. Assuming $1 per hour for an M60 GPU (average price across cloud providers), it would cost $7,200 to render 100,000 images. Though this seems expensive, real data collection costs can run much higher, especially if we take annotation into consideration.

Acknowledgements We thank Pedro Urbina, Jon Hanzelka, Rodney Brunet, and Panagiotis Giannakopoulos for their artistic contributions; and Virginia Estellers and Matthew Johnson for contributions to the face model.

References

[1] Amazon. Amazon climate pledge. https://www.aboutamazon.com/planet/climate-pledge, 2021. 8

[2] G. R. Anderson, M. J. Aftosmis, and M. Nemec. Parametric Deformation of Discrete Geometry for Aerodynamic Shape Design. *Journal of Aircraft*, 2012. 5

[3] S. Bak, P. Carr, and J.-F. Lalonde. Domain Adaptation through Synthesis for Unsupervised Person Re-identification. In *ECCV*, 2018. 3

[4] T. Baltrušaitis, P. Robinson, and L.-P. Morency. 3D Constrained Local Model for Rigid and Non-Rigid Facial Tracking. In *CVPR*, 2012. 2

[5] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999. 2

[6] Blender Foundation. Cycles renderer. https://www.cycles-renderer.org/, 2021. 5

[7] B. Browatzki and C. Wallraven. 3FabRec: Fast Few-shot Face alignment by Reconstruction. In *CVPR*, 2020. 7

[8] M. J.-Y. Chiang, B. Bitterli, C. Tappan, and B. Burley. A practical and controllable hair and fur model for production path tracing. In *Computer Graphics Forum*, 2016. 5

[9] P. H. Christensen. An approximate reflectance profile for efficient subsurface scattering. In *SIGGRAPH Talks*, 2015. 4

[10] CLO Virtual Fashion Inc. Marvelous designer. https://www.marvelousdesigner.com/, 2021. 5

[11] P. Debevec. Image-based lighting. In *SIGGRAPH Courses*, 2006. 5

[12] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. VirtualWorlds as Proxy for Multi-object Tracking Analysis. In *CVPR*, 2016. 2

[13] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *JMLR*, 2016. 1, 3

[14] S. J. Garbin, Y. Shen, I. Schuetz, R. Cavin, G. Hughes, and S. S. Talathi. OpenEDS: Open eye dataset. *arXiv preprint arXiv:1905.03702*, 2019. 8

[15] S. J. Garbin, M. Kowalski, M. Johnson, and J. Shotton. High resolution zero-shot domain adaptation of synthetically rendered face images. In *ECCV*, 2020. 3

[16] B. Gecer, B. Bhattarai, J. Kittler, and T.-K. Kim. Semi-supervised Adversarial Learning to Generate Photorealistic Face Images of New Identities from 3D Morphable Model. In *ECCV*, 2018. 3

[17] T. Gerig, A. Forster, C. Blumer, B. Egger, M. Lüthi, S. Schönborn, and T. Vetter. Morphable face models - an open framework. *Automatic Face and Gesture Recognition*, 2017. 3, 4

[18] Google. Google cloud sustainability. https://cloud.google.com/sustainability/, 2021. 8

[19] T. Guo, Y. Kim, H. Zhang, D. Qian, B. Yoo, J. Xu, D. Zou, J.-J. Han, and C. Choi. Residual encoder decoder network and adaptive prior for face parsing. In *AAAI*, 2018. 7

[20] R. A. Güler, G. Trigeorgis, E. Antonakos, P. Snape, S. Zafeiriou, and I. Kokkinos. DenseReg: Fully Convolutional Dense Shape Regression In-the-Wild, 2017. 7

[21] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *CVPR*, 2016. 6

[22] D. Hendler, L. Moser, R. Battulwar, D. Corral, P. Cramer, R. Miller, R. Cloudsdale, and D. Roble. Avengers: Capturing Thanos's Complex Face. In *SIGGRAPH Talks*, 2018. 1

[23] T. Hodaň, V. Vineet, R. Gal, E. Shalev, J. Hanzelka, T. Connell, P. Urbina, S. N. Sinha, and B. Guenter. Photorealistic image synthesis for object instance detection. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 66–70. IEEE, 2019. 2

[24] L. A. Jeni, J. F. Cohn, and T. Kanade. Dense 3D face alignment from 2D videos in real-time. In *11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, 2015. 2

[25] A. Kar, A. Prakash, M.-Y. Liu, E. Cameracci, J. Yuan, M. Rusiniak, D. Acuna, A. Torralba, and S. Fidler. Meta-Sim: Learning to Generate Synthetic Datasets. In *ICCV*, 2019. 2, 3

[26] B. Karis, T. Antoniades, S. Caulkin, and V. Mastilovic. Digital humans: Crossing the uncanny valley in ue4. Game Developers Conference, 2016. 1

[27] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of StyleGAN. In *CVPR*, 2020. 1

[28] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[29] M. Kowalski, S. J. Garbin, V. Estellers, T. Baltrušaitis, M. Johnson, and J. Shotton. Config: Controllable neural face image generation. In *ECCV*, 2020. 3

[30] F. Kuhnke and J. Ostermann. Deep head pose estimation using synthetic images and partial adversarial domain adaption for continuous label spaces. In *CVPR*, 2019. 2

[31] K. Kärkkäinen and J. Joo. Fairface: Face attribute dataset for balanced race, gender, and age. In *WACV*, 2021. 1

[32] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. S. Huang. Interactive facial feature localization. In *ECCV*, 2012. 6

[33] J. P. Lewis, M. Cordner, and N. Fong. Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation. In *SIGGRAPH*, 2000. 3

[34] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. Learning a model of facial shape and expression from 4D scans. *SIGGRAPH Asia*, 2017. 3

[35] J. Lin, H. Yang, D. Chen, M. Zeng, F. Wen, and L. Yuan. Face Parsing with RoI Tanh-Warping. In *CVPR*, 2019. 6, 7

[36] Y. Liu, H. Shi, H. Shen, Y. Si, X. Wang, and T. Mei. A new dataset and boundary-attention semantic segmentation for face parsing. In *AAAI*, 2020. 6, 7

[37] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *SIGGRAPH Asia*, 2015. 5

[38] I. Lozano, J. Saunier, S. Panhard, and G. Loussouarn. The diversity of the human hair colour assessed by visual scales and instrumental measurements. a worldwide survey. *International journal of cosmetic science*, 39:101–107, 2017. 5

[39] Microsoft. Microsoft will be carbon negative by 2030. https://blogs.microsoft.com/blog/2020/01/16/microsoft-will-be-carbon-negative-by-2030/,

2021. 8

[40] F. Mueller, F. Bernard, O. Sotnychenko, D. Mehta, S. Sridhar, D. Casas, and C. Theobalt. GANerated Hands for Real-Time 3D Hand Tracking from Monocular RGB. In *CVPR*, 2018. 2, 3

[41] H. Ning, W. Xu, Y. Gong, and T. Huang. Discriminative learning of visual words for 3d human pose estimation. In *CVPR*, 2003. 2

[42] B. Nojavanasghari, T. Baltrušaitis, C. E. Hughes, , and L.-P. Morency. Hand2face: Automatic synthesis and recognition of hand over face occlusions. In *ACII*, 2017. 2

[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *NeurIPS*, 2019. 6

[44] W. Qiu, F. Zhong, Y. Zhang, S. Qiao, Z. Xiao, T. S. Kim, Y. Wang, and A. Yuille. Unrealcv: Virtual worlds for computer vision. *ACM Multimedia Open Source Software Competition*, 2017. 2

[45] E. Richardson, M. Sela, and R. Kimmel. 3d face reconstruction by learning from synthetic data. *International Conference on 3D Vision*, 2016. 2, 4

[46] E. Richardson, M. Sela, R. Or-El, and R. Kimmel. Learning detailed face reconstruction from a single image. In *CVPR*, 2017. 2

[47] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for data: Ground truth from computer games. In *European conference on computer vision*, pages 102–118. Springer, 2016. 2

[48] J. P. Robinson, Y. Li, N. Zhang, Y. Fu, , and S. Tulyakov. Laplace Landmark Localization. In *ICCV*, 2019. 7

[49] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, 2015. 6

[50] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes. In *CVPR*, 2016. 2

[51] A. Rozantsev, V. Lepetit, and P. Fua. On rendering synthetic images for training an object detector. *CVIU*, 2014. 2

[52] Russian3DScanner. Wrap3. `https://www.russian3dscanner.com/`, 2021. 4

[53] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. 300 faces In-the-wild challenge: Database and results. *Image and Vision Computing (IMAVIS)*, 2016. 6, 7

[54] S. Saito, Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li. PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization. In *ICCV*, October 2019. 2

[55] S. Saito, T. Simon, J. Saragih, and H. Joo. PIFuHD: Multi-Level Pixel-Aligned Implicit Function for High-Resolution 3D Human Digitization. In *CVPR*, 2020. 2, 4

[56] F. S. Saleh, M. Sadegh Aliakbarian, M. Salzmann, L. Petersson, and J. M. Alvarez. Effective Use of Synthetic Data for Urban Scene Semantic Segmentation. In *ECCV*, 2018. 3

[57] S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa. Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation. In *CVPR*, 2018. 3

[58] M. Sela, E. Richardson, and R. Kimmel. Unrestricted facial geometry reconstruction using image-to-image translation. *ICCV*, 2017. 2

[59] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011. 2

[60] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017. 1, 3

[61] T. Simon, H. Joo, I. Matthews, and Y. Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017. 2

[62] Y. Sugano, Y. Matsushita, and Y. Sato. Learning-by-Synthesis for Appearance-Based 3D Gaze Estimation. In *CVPR*, 2014. 2

[63] L. Świrski and N. A. Dodgson. Rendering synthetic ground truth images for eye tracker evaluation. In *ETRA*, 2014. 2

[64] G. Te, Y. Liu, W. Hu, H. Shi, and T. Mei. Edge-aware Graph Representation Learning and Reasoning for Face Parsing. In *ECCV*, 2020. 7

[65] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid. Learning from synthetic humans. In *CVPR*, 2017. 2

[66] X. Wang, L. Bo, and L. Fuxin. Adaptive Wing Loss for Robust Face Alignment via Heatmap Regression. In *ICCV*, 2010. 7

[67] Z. Wei, S. Liu, Y. Sun, and H. Ling. Accurate facial image parsing at real-time speed. *IEEE Transactions on Image Processing*, 28(9):4659–4670, 2019. 7

[68] E. Wood, T. Baltrušaitis, X. Zhang, Y. Sugano, P. Robinson, and A. Bulling. Rendering of eyes for eye-shape registration and gaze estimation. In *ICCV*, 2015. 1, 2

[69] E. Wood, T. Baltrušaitis, L.-P. Morency, P. Robinson, and A. Bulling. Learning an appearance-based gaze estimator from one million synthesised images. In *ETRA*, 2016. 2

[70] W. Wu, C. Qian, S. Yang, Q. Wang, Y. Cai, and Q. Zhou. Look at Boundary: A Boundary-Aware Face Alignment Algorithm. In *CVPR*, 2018. 7

[71] W. Wu, N. Lu, and E. Xie. Synthetic-to-real unsupervised domain adaptation for scene text detection in the wild. In *ACCV*, 2020. 3

[72] P. Yakubovskiy. Segmentation models pytorch. `https://github.com/qubvel/segmentation_models.pytorch`, 2020. 6

[73] Y. Yao, L. Zheng, X. Yang, M. Naphade, and T. Gedeon. Simulating Content Consistent Vehicle Datasets with Attribute Descent. In *ECCV*, 2020. 2

[74] L. Yin, X. Chen, Y. Sun, T. Worm, and M. Reale. A high-resolution 3d dynamic facial expression database. In *2008 8th IEEE International Conference on Automatic Face Gesture Recognition*, pages 1–6, 2008. doi: 10.1109/AFGR.2008.4813324. 2

[75] G. Zaal, S. Majboroda, and A. Mischok. HDRI haven. `https://hdrihaven.com`, 2020. 5

[76] X. Zeng, X. Peng, and Y. Qiao. Df2net: A dense-fine-finer

network for detailed 3d face reconstruction. In *CVPR*, 2019. 2, 3, 4

[77] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *ICCV*, 2017. 3

[78] M. Zhu, D. Shi, M. Zheng, and M. Sadiq. Robust Facial Landmark Detection via Occlusion-Adaptive Deep Networks. In *CVPR*, 2019. 7

[79] X. Zhu, X. Liu, Z. Lei, and S. Z. Li. Face alignment in full pose range: A 3d total solution. *TPAMI*, 2017. 2