3D Morphable Face Models - Past, Present and Future

To cite this version:

HAL Id: hal-02280281
https://inria.hal.science/hal-02280281v2
Submitted on 3 Sep 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
1 INTRODUCTION

It is 20 years since 3D Morphable Face Models were first presented at SIGGRAPH ’99. They were proposed as a general face representation and a principled approach to image analysis. Blanz and Vetter [1999] introduced and tackled many subsidiary problems and the results were considered groundbreaking. The impact of the original paper has been long term, recognized by an impact paper award, and the approach and applications are accessible to a wide audience (the original supplementary video was one of the most popular videos in the early days of YouTube). However, the approach is not just of historical interest. In the past two years, 3D Morphable Face Models have been re-discovered in the context of deep learning and are incorporated into many state-of-the-art solutions for face analysis.

This survey aims to build a starting point for researchers new to the topic, act as a reference guide for the community around 3D Morphable Models and to introduce exciting open research questions.

1.1 Definition

A 3D Morphable Face Model is a generative model for face shape and appearance that is based on two key ideas: First, all faces are in dense point-to-point correspondence, which is usually established on a set of example faces in a registration procedure and then maintained throughout any further processing steps. Due to this
correspondence, linear combinations of faces may be defined in a meaningful way, producing morphologically realistic faces (morphs). The second idea is to separate facial shape and color and to disentangle these from external factors such as illumination and camera parameters. The Morphable Model may involve a statistical model of the distribution of faces, which was a principal component analysis in the original work [Blanz and Vetter 1999] and has included other learning techniques in subsequent work.

1.2 History

The initial research question behind the idea of 3D Morphable Models (3DMM) was how a visual system, biological or artificial, can cope with the high variety of images that a single class of objects can generate, and how objects are represented to solve vision tasks. The leading assumption for the development of 3DMMs was that prior knowledge about object classes plays an important role in vision and helps to solve otherwise ill-posed problems. 3DMMs are designed to capture such prior knowledge, and they are learned automatically from a set of examples. The representation is general, so it may be applied to different objects and tasks.

Representations of faces and the task of face recognition have been in the focus of vision research for a long time. An important and very influential paradigm shift in this field was the Eigenfaces approach by Sirovich and Kirby [1987] and Turk and Pentland [1991], which learned an explicit face representation from examples and operated entirely on grey-levels in the image domain. Eigenfaces treated images of faces as a vector space and performed a principal component analysis, with the eigenvectors representing the main modes of variation in that space. The drawback of Eigenfaces was not only that it was limited to a fixed pose and illumination, but that it had no effective representation of shape differences: When the coefficients in linear combinations of eigenvectors are changed continuously, structures will fade in and out, rather than shift along the image plane. As a consequence, the model fails to find a single parameter for, say, the distances between the eyes. The Eigenfaces approach was also extended to 3D face surfaces by Atick et al. [1996] to model shading variations in faces, yet with essentially the same limitation.

Several research groups proceeded by adding an Eigendecomposition of 2D shape variations between individual faces. This provided both an explicit shape model, and - after warping the images - an aligned Eigenface model without blurring and ghosting artifacts. While in the original Eigenface approach, the images were only aligned by a single point (e.g., the tip of the nose), the new methods established correspondence on significantly more points. Landmark-based face warping for image analysis was introduced by Craw and Cameron [1991]. Using approximately 200 landmarks, the first statistical shape model was proposed in Active Shape Models [Cootes et al. 1995]. While this model used shape only, Active Appearance Models [Cootes et al. 1998] proposed a combination of shape and appearance that turned out to be very successful and influential. Other groups computed dense pixel-wise image correspondences with optic-flow algorithms for modeling the facial shape variations [Hallinan et al. 1999; Jones and Poggio 1998]. In all these correspondence-based approaches, images are warped to a common template, and the appearance variation is then performed in the same way as the original Eigenfaces, but on the shape-normalized images. The shape model, on the other hand, provides a powerful and compact representation of shape differences by shifting pixels in the image plane. However, compared to the simple linear projection in Eigenfaces, the image analysis task is transformed into a more challenging nonlinear model fitting problem.

These 2D models were efficient to cover the shape variation for a fixed pose and illumination setting. The framework was extended to variations across pose by Vetter and Poggio [1997] and to other object classes, such as images of cars [Jones and Poggio 1998]. All this groundwork demonstrated that a separation of shape and texture information in images can model the variation of faces. On the other hand, the price to pay for taking pose and illumination variations into account was high: eventually, it would require many separate models, each limited to a small range of poses and illuminations. In contrast, the progress of 3D Computer Graphics in the 1990s demonstrated that variations in pose and illumination are easy to simulate, including self-occlusion and shadowing. Adapting methods from graphics to face modeling and computer vision led to the new face representation in 3DMMs and the idea of using analysis-by-synthesis to map between the 3D and 2D domain. Those were the two key contributions in the first paper on 3DMMs [Blanz and Vetter 1999], compare Figure 2. The name Morphable Model was derived from their 2D counterpart [Jones and Poggio 1998], and in fact, Jones and Poggio strongly influenced the ideas that led to 3DMMs.

3DMMs and 2D Morphable Models rely on dense correspondence, rather than only a set of facial feature points. In the original work, this was established by an optical flow algorithm for image registration. The image synthesis algorithm used a standard rendering model with perspective projection, ambient and directional lighting, and a Phong model of surface reflectance that includes a specular component. However, in analysis-by-synthesis, this approach comes at a computational price because shape-camera [Smith 2016] and illumination-albedo [Egger 2018] ambiguities lead to a hard ill-posed optimization problem. Moreover, the optimization is costly and is prone to end in unwanted local optima. Just as it is already dramatically more complicated to fit an Active Appearance Model
to a 2D image, compared to the simple projection needed for Eigenfaces, the complexity of 3DMM fitting raises additional problems which have remained challenging to researchers after 20 years of development.

At the time the initial 3DMM was developed, image-based models were dominating computer vision and even animation [Ezzat et al. 2002], and they were rather elaborate at that time. It was a key decision to take the best of both the 2D and the 3D world, by using 3D models to manipulate existing images on the one hand, and applying 2D algorithms to 3D surfaces: Unlike mesh-based algorithms, the original 3DMM used optical flow, multi-resolution approaches and interpolation algorithms on parameterized surfaces of faces. With the initial face scanner delivering surfaces in a two-dimensional cylinder parameterization, all those steps were performed in 2D, and most of the methods involved were replaced with their 3D equivalent only many years later. It is interesting to see that after a development towards 3D, the computer vision community came back to 2D representations by using deep learning, and now evolves again to 3D, e.g., by integrating 3DMMs.

Over the past years, 3DMMs were applied beyond faces. Models were built for the surface of the human body [Allen et al. 2003; Anguelov et al. 2005; Loper et al. 2015] and for other specific parts of the body like ears [Dai et al. 2018] and hands [Khamis et al. 2015], animals [Sun and Murata 2020; Zuffi et al. 2018] and even cars [Shelton 2000]. In this survey, we focus on 3DMMs to model the human face, though many of the techniques and challenges are the same across different object classes.

The 3DMM was developed in a time where algorithms and data were rarely shared across researchers and institutions. 10 years later the first publicly available 3DMM was released [Paysan et al. 2009a] and in the last 10 years, all individual data and algorithmic components needed to build and use 3DMMs were released by various researchers. We collected a list of all available resources and will further maintain it [Community 2019].

The 3DMM was built as a general representation for faces, not just aiming at one specific task. Even though the model is outperformed for some very specific applications such as face recognition, it is unique in its generality across different tasks and applications.

1.3 Organization

There is a recent state of the art report on monocular 3D reconstruction, tracking and applications [Zollhoefer et al. 2018]. This focuses on the most recent advances, particularly related to the specific task of tracking and reconstruction. In contrast, in this paper, we instead focus on the 3DMM, all involved methods, and reflect the major contributions over the past 20 years while at the same time highlighting challenges and future directions.

This survey is organized from building to applying a 3DMM. We start with Section 2 where we present methods to acquire 3D facial data for model building. We then describe in Section 3 the various approaches to model the 3D shape and facial appearance. In Section 4 we discuss the methods to generate a 2D image from our 3D model using computer graphics. Our Section 5 surveys the major application of 3DMMs, namely the reconstruction of a 3D face from a 2D image. Section 6 summarizes the impact of 3DMMs in the recent advances in the field of deep learning and how deep learning can be used to improve the modeling and analysis. Section 7 summarizes the various applications where 3DMMs were used in the past 20 years. Every Section summarizes the major challenges the authors see regarding the current limitations of 3DMM. We also collect challenges that are shared across multiple Sections in Section 8, where we also venture an outlook on what we expect to see in the next 10 to 20 years and how the 3DMM will keep impacting how faces are represented.

2 FACE CAPTURE

The key ingredient to any 3DMM is a representative set of 3D shapes, usually coupled with corresponding appearance data. The typical way to construct such a sample pool is by acquiring data from the real world. In this section, we give a brief overview of different approaches that have been used to acquire facial data as well as data of facial parts. As we are concerned with the creation of input datasets for 3DMMs, we limit the discussion to acquisition under controlled conditions, as opposed to the more challenging in-the-wild setting. Note that controlled 3D face capture may not always be necessary. There have been attempts to learn 3DMMs directly from images [Cashman and Fitzgibbon 2012] and state-of-the-art deep learning-based methods simultaneously learn a 3DMM and regression-based fitting from 2D training data (see Section 6.3). In this section we begin by covering shape acquisition methods in Section 2.1 including geometric, photometric and hybrid methods. Sections 2.2, 2.3 and 2.4 describe methods for capture of appearance, face parts and dynamics respectively. Section 2.5 lists publicly available 3D face datasets that could be exploited for building 3DMMs. Finally, we consider open challenges related to face capture in Section 2.6.

2.1 Shape Acquisition

The three-dimensional shape is arguably the most important ingredient to a 3DMM. The issue of shape representation has not been widely considered in the context of 3DMMs. By far the most commonly used representation is a triangle mesh. Rare exceptions include cylindrical [Atick et al. 1996] and orthographic [Dovgaid and Basri 2004] depth maps (though these representations do not permit meaningful dense correspondence), per-vertex surface normals [Aldrian and Smith 2012], and, more recently, volumetric orientation fields [Saito et al. 2018] and signed distance functions [Park et al. 2019]. Using a triangle mesh representation, dense correspondence requires that all samples exhibit the same topology and that the vertices encode the same semantic point on all samples. Establishing correspondence across the samples is a challenging topic in itself, discussed in Section 3.5. In this section, we focus on the acquisition of raw 3D data, before establishing correspondence.

2.1.1 Geometric methods

Geometric methods estimate directly the 3D coordinates of a shape either by observing the same surface point from two or more viewpoints (in which case the challenge is identifying corresponding points between images) or by observing a projected pattern (in which case the challenge is identifying the correspondence between the known pattern and an image of its projection). Methods can either be considered active, i.e. they emit light or other signals into the scene, or passive. Laser scanners,
Time-of-Flight sensors, and Structured Light systems are active systems, where multi-view photogrammetry is a passive alternative. Active multi-view photogrammetry may be considered a hybrid active/passive approach, as it relies on passive photogrammetry to reconstruct the shape, but augments the object with a well-defined texture projection that benefits the reconstruction [Zhang et al. 2004]. Unlike structured light, the origin of the light does not matter as the projected texture is solely meant to augment the texture used for multi-view stereo matching. This type of technology is used by the Intel® RealSense™ D435 camera for example1. In the early days of 3DMMs, active systems were the only real option to acquire 3D shapes at a reasonable quality. The original paper of Blanz and Vetter [1999] relied on laser scanning [Levoy et al. 2000], where the face is rasterized via one or more laser-beams. The laser beam illuminates the face surface at a point and using the known camera/laser arrangement the 3D position of this point may be triangulated. The biggest drawback of laser scanners is the acquisition time, as only very few samples are gathered at any given time – even at very high frame rates, such systems require the subjects to sit still for several seconds.

Structured light scanners [Geng 2011] overcome this limitation to some extent by injecting not only a few beams but leveraging projectors that offer millions of them. The challenge here is to identify which beam is illuminating the object at a given point. This is addressed by structuring the projected light in a way that allows to clearly identify the origin of any ray. The simplest approach is binary encoding, which projects black and white patterns assigning a unique binary code to each pixel. The required number of patterns is still quite substantial, for VGA resolution one needs 19 distinct patterns and for 4K resolution 23 patterns, and hence this approach is most suited for capturing static objects. However, technical improvements have begun to make these approaches viable for dynamic capture of faces. The Intel® RealSense™ SR300 uses only 9 binary patterns to obtain VGA, while the most recent RealSense depth camera produces VGA resolution at 60 depth FPS with a scanning laser technology. Other more complex structured light methods have been proposed, such as gray codes or (colored) fringe patterns, which can reduce the number of required frames further, in extreme cases even to a single frame. A very popular commercial system that was used to create face datasets ([Cao et al. 2014b]) and that employs structured light is the first generation Kinect sensor2. The device employs a structured dot pattern, which allows reconstructing depth from a single frame by sacrificing spatial resolution. Resolution may be improved by accumulating several frames [Newcombe et al. 2011]. With the increased resolution and quality of consumer cameras, passive systems have become the method of choice in most cases, since they are simpler to assemble and operate and off-the-shelf photogrammetry software solutions, both commercial such as Agisoft3 or RealityCapture4, as well as open-source solutions such as Meshroom5, provide very good results on human faces. Also, complete systems can be purchased that come with both hard- and software678. These methods typically do not require the aggregation of information over time and hence offer themselves for single-shot acquisition [Beeler et al. 2010] as well as full-frame rate performance capture [Beeler et al. 2011; Bradley et al. 2010; Furukawa and Ponce 2009]. A potential disadvantage of the aforementioned systems is their form factor since they all require at least some separation between the different participating components, i.e. the cameras or lights, often referred to as the baseline. An alternative which becomes more and more viable due to the push of the mobile industry are time-of-flight sensors, where the elements can be located close to each other. The second-generation Kinect

1https://www.intelrealsense.com/depth-camera-d435/
3https://www.agisoft.com
4https://www.capturingreality.com
5https://alicevision.org/
6https://www.canfieldsci.com/imaging-systems/vectra-m3-3d-imaging-system/
7http://www.dvid.com
8http://www.3dmd.com/
sensor\(^9\) belongs to this family, as well as many depth sensors that are shipped with modern mobile phones. A challenge that time-of-flight sensors share with most of the active systems is that color information has to be acquired separately and is not intrinsically aligned with the 3D data, which is another advantage of passive setups.

2.1.2 Photometric methods. Photometric methods typically estimate surface orientation, from which the 3D shape may be recovered via integration. The challenge here is to select models that accurately capture reflectance properties of the surface and obtaining sufficient measurements that the inversion of these models is well-posed. Compared to geometric methods, photometric methods typically offer higher shape detail and do not rely on the presence of matchable features (so are applicable to smooth, featureless surfaces) but often suffer from low-frequency bias in the reconstructed positions caused by modeling errors in reflectance and illumination. Photometric stereo \(\text{[Ackermann et al. 2015]}\) estimates the surface normal at each pixel by observing a scene from a fixed position under at least three different illumination conditions, which can be spectrally multiplexed \(\text{[Hernández et al. 2007]}\) in order to reduce the number of frames required. Early work assumed known lighting directions and perfectly diffuse reflectance. When illumination is uncalibrated and a more suitable glossy reflectance model is used, generic face priors can be used to resolve the resulting ambiguity \(\text{[Georgiades 2003]}\). Typically more lighting conditions are used to increase robustness and coverage, such as four \(\text{[Zafeiriou et al. 2013]}\) or even nine \(\text{[Gotardo et al. 2015]}\). Gradient-based illumination takes the number of conditions to the extreme, by illuminating the subject not with discrete individual point lights, but by an ideally continuous, omnidirectional incident illumination gradient. An advantage of this set up is that hard light source occlusions (cast shadows) are replaced by soft partial occlusions of the illuminating hemisphere (ambient occlusion). In practice, the omnidirectional illumination is realized via a light-stage \(\text{[Debevec et al. 2000]}\), which discretizes the gradient with a large number (several hundred) of light sources. The original work of Ma et al. \(\text{[2007]}\) suggests the use of four distinct gradients, which has later been extended using complementary gradients \(\text{[Wilson et al. 2010]}\). Again, variants of temporal, spectral and polarization multiplexing have been proposed to reduce the number of required conditions.

2.1.3 Hybrid methods. Hybrid methods combine the strength of geometric and photometric methods, specifically, they reduce the low-frequency bias typically present in photometric methods and increase the high-frequency details when compared to geometric methods. Nehab et al. \(\text{[2005]}\) propose a method for merging the low frequencies of positional information and the high frequencies of surface normals. The method is particularly efficient, involving only the solution of a sparse linear system of equations, and has been used in the context of 3DMM fitting \(\text{[Patel and Smith 2012]}\). Various combinations of geometric and photometric methods have been considered. For example, Zivanov et al. \(\text{[2009]}\) combine structured light with photometric stereo, Ma et al. \(\text{[2007]}\) combine structured light with gradient-based illumination, Ghosh et al. \(\text{[2011]}\) combine multi-view stereo with gradient-based illumination, and Beeler et al. \(\text{[2010]}\) combine passive multi-view photogrammetry with shape-from-shading. Figure 3(d) shows the output of a hybrid method in which photometric surface normals are merged with a multi-view stereo mesh.

2.2 Appearance Capture

In addition to shape, appearance is also required for many 3DMM tasks, such as synthesizing images (see Section 4) and inverse rendering (see Section 5). Unlike shapes, which are almost exclusively represented as triangular meshes, appearance representation varies substantially. While in theory, every vertex of the mesh could have an associated appearance property, typically shapes are parameterized to the 2D domain and textures are used to store appearance properties. Appearance can be as simple as backprojecting the color of the images onto the shapes, which causes shading effects to be baked in. Self-occlusion, in particular when only a single viewpoint is available, results in missing data in the occluded areas, which must be hallucinated somehow. Booth et al. \(\text{[2018b]}\) use 3DMM fits to in-the-wild images and Principal Component Pursuit with missing values to complete the unobserved texture. They build their appearance model directly on the sampled textures. Such a simplistic approach, however, does not allow intrinsic face appearance properties to be separated from shading/shadowing (and hence illumination/geometry). A partial solution to this problem is to control illumination conditions during capture, for example by using multiple light sources to create approximately ambient lighting. Note that a truly Lambertian convex surface observed under truly ambient light gives exactly the albedo \(\text{[Lee et al. 2005]}\). The appearance models in the most popular 3DMMs \(\text{[Booth et al. 2018a; Dai et al. 2017; Paysan et al. 2009a]}\) use this approach, combining images from multiple cameras to provide full coverage of the face with diffuse lighting to approximate albedo. A better approach is to explicitly separate shading from skin color, often referred to as intrinsic decomposition. This allows relighting of the face under novel incident illumination conditions and a 3DMM built on such data truly models intrinsic characteristics of the face. Several approaches have been presented over the years to acquire reflectance data suited for parametric rendering, measuring surface reflectance \(\text{[Marschner et al. 1999]}\) and even subsurface scattering properties \(\text{[Ghosh et al. 2008]}\). The polarised spherical illumination environment used by Ma et al. \(\text{[2007]}\) enables diffuse albedo to be captured in a single shot and specular albedo in two images (see Figure 3(a) and (b)). While such approaches have predominately used active setups, recently capture under passive conditions has been demonstrated \(\text{[Gotardo et al. 2018]}\).

2.3 Face part specific methods

Certain parts of the human face require more targeted acquisition methods and devices since they do not conform with the assumptions typically made by abovementioned approaches. For example, the frontmost part of the eye, the cornea, is for obvious reasons fully transparent and distorts the appearance of the underlying iris due to refraction. Bérard et al. \(\text{[2014]}\) leverage a combination of several specialized algorithms, including shape-from-specularity,
in order to reconstruct all visible components of the eye. Another challenging example are teeth [Wu et al. 2016a], which exhibit extremely challenging appearance [Velinov et al. 2018]. Hair violates the common assumption that the reconstructed shape is a smooth continuous surface, and requires specialized approaches that estimate hair fibers [Beeler et al. 2012], hair strands [Hu et al. 2014a; Luo et al. 2013] and braiding [Hu et al. 2014b], or even encode hair as a surface [Echevarria et al. 2014] for manufacturing. While most hair acquisition focuses on static reconstruction, some do capture hair in motion [Xu et al. 2014] or estimate physical properties for hair simulation [Hu et al. 2017a]. Especially challenging is the acquisition of partially or completely hidden properties, such as the tongue [Hewer et al. 2018], the skull [Achenbach et al. 2018; Beeler and Bradley 2014], or the jaw [Zoss et al. 2019, 2018], where oftentimes specialized imaging systems are required, such as Computer Tomography (CT), Magnetic Resonance Imaging (MRI), or Electromagnetic Articulography (EMMA). Lastly, even skin itself requires specialized treatment in some areas, such as lips [Garrido et al. 2016b] or eyelids [Bermano et al. 2015], where the local appearance and deformation exceed the capabilities of the more generic methods.

2.4 Dynamic capture

Historically, 3DMMs have been mostly concerned with static shapes, for example with a set of neutral shapes from different individuals or with a discrete set of expressions per individual, neglecting how the face transitions between expressions. Most capture systems used to build 3DMMs were hence static systems, focused on capturing individual shapes rather than full performances. As the field begins to integrate more temporal information into the models, the need for dynamic capture systems will rise. Active systems have been considered, both geometric [Zhang et al. 2004] and photometric [Wilson et al. 2010]. However, passive systems [Beeler et al. 2011; Bradley et al. 2010] are currently the technologies of choice, since they do not require temporal multiplexing and still deliver high-quality shapes, and more recently even per-frame reflectance data [Gotardo et al. 2018]. A beneficial side-effect of such technologies is that they often provide shapes that are already in correspondence, removing the need to establish correspondence in a post-processing step (Section 3.5), and making them attractive solutions even when only a discrete set of shapes is desired. Available commercial solutions include Di4D10, 3dMD11, or the Medusa system12.

2.5 Publicly available face datasets

A relatively large number of publicly available datasets exist that could be leveraged in the construction of 3DMMs, though many have never been used for this purpose. We believe there is not broad awareness of the range of 3D datasets available and so collect them together in Table 1. We hope that this will encourage work that seeks to exploit multiple datasets for 3DMM building.

2.6 Open challenges

The field of face capture is far ahead of face modeling in general and 3DMMs in particular. There is a large gap between the quality of data that can be captured and the data actually used to build 3DMMs. There is a further gap between the quality of this already-deficient data and what a 3DMM is able to synthesize (see Section 3). Hence, from the perspective of 3DMMs, the open challenges in capture do not generally relate to improving the acquisition quality, but to the lack of publicly available data. While there is a decent number of datasets publicly available (see Section 2.5), most of these contain only moderate quality shape data and no appearance information, with the exception of [Stratou et al. 2011], which consists of 23 identities only. We believe that the lack of high-quality datasets is due to a variety of reasons. On the one hand, high-quality acquisition devices that can capture both shape and appearance are not readily available. Most of them are custom-built, cannot easily be purchased or licensed, and require expert knowledge for operation. On the other hand, acquiring and processing data may be a time and resource-intensive effort, since many systems in the research community were not conceived for scalable deployment but for experimental use; slow capture methods are not applicable to young or elderly people, expensive setups are challenging to replicate on a global scale to capture whole populations, and methods requiring very bright illumination makes it unpleasant to be captured with eyes open. Furthermore, most high-quality systems, in particular ones that also measure appearance, generally require controlled lab conditions which makes it difficult to capture large numbers of the general public. Advances in face capture may alleviate some of these issues.

Additionally, there are many important broader questions related to data acquisition that remain unanswered. How many faces do we really need to capture in order to build a representative (universal) model? How can we ensure we capture natural expressions? Most people are not trained to perform specific expressions (i.e. FACS13), and will have difficulties performing normally when put in a capture setup, leading to a biased dataset. How should we deal with bias in general and what is the right sampling strategy with respect to age, gender, ethnicity and so on? Are the capture methods themselves biased? For example, capturing faces with very dark skin is challenging for both photometric and geometric methods. Should we accept that we cannot hope to capture sufficiently broad data and therefore rely on synthesizing additional data or using captured data to build a bootstrap model that is refined on large 2D datasets? These approaches are discussed in Section 6.

Finally, there are some philosophical and ethical issues to consider. The human face is unique and highly personal. Once a face has been captured in high detail, it is possible to synthesize new images that are almost indistinguishable from photos. If captured datasets are made publicly available, it is very difficult to control the distribution and use of such data. Obtaining proper informed consent is, therefore, both legally and ethically important but perhaps even this does not go far enough, particularly when consent for minors is given by parents. These issues are beyond the expertise

---

10http://www.di4d.com/
11http://www.3dmd.com/
12https://studios.disneyresearch.com/medusa/
13https://en.wikipedia.org/wiki/Facial_Action_Coding_System
of computer graphics and vision researchers and perhaps suggest a need for discussion and debate with other disciplines.
3 MODELING

This section outlines how to compute a 3DMM by modeling the variations of digitized 3D human faces. In particular, the following three types of variations are commonly considered. First, geometric variations across different identities are captured in a shape model, as outlined in Section 3.1. Commonly used models include global models, which represent variations of the entire face surface, and local models, which represent variations of facial parts. Second, geometric variations across different facial expressions are captured in an expression model, as outlined in Section 3.2. Commonly used models can be mainly classified into additive and multiplicative models. More recently, nonlinear expression models are starting to be explored. Third, variation in appearance and illumination are captured in a separate appearance model as outlined in Section 3.3.

It is interesting to note that the landmark paper on 3DMMs published 20 years ago [Blanz and Vetter 1999] proposed first models for all three types of variation that are still commonly used today.

To compute shape, expression, or appearance models, statistics are performed over a database of face data, where traditionally 3D scans of faces were used, and more recently some approaches also learn face models directly from 2D images, as outlined in Section 6.3. This computation of statistics requires correspondence information, that is, anatomically corresponding parts of the faces need to be compared, and hence known either explicitly or implicitly. An overview of how correspondence information is computed for faces is given in Section 3.5. The most commonly used approach is to compute correspondence information explicitly before computing the 3DMM. Some recent methods compute correspondence information at the same time while the 3DMM is built.

3DMMs are generative models and the ability to synthesize novel faces is a key feature and briefly discussed in Section 3.6. Finally, this section provides a list of available models and discusses open challenges on 3D face modeling in Sections 3.7 and 3.8, respectively.

3.1 Shape models

This section considers modeling geometric variation across different subjects computed using classical modelling approaches that use 3D data. To use a set of 3D scans as training data, we require a distance measure between any pair of scans, and computing a distance between raw scans consisting of different numbers of unstructured vertices is a complex problem. Most commonly, the community proceeds by first pre-processing the dataset by deforming a template mesh to all scans, which establishes anatomic correspondences between the points of the scans (see Section 3.5). We denote the surface of such a pre-processed mesh by \( S \) in the following. The \( i \)-th vertex of \( S \) is denoted by \( v_i \in \mathbb{R}^3 \), and its associated vector \( c_i \in \mathbb{R}^n \) contains the coordinates of \( v_i \) in a fixed order. All meshes share a common triangulation. We denote the \( i \)-th triangle by \( t_i = (i_1^i, i_2^i, i_3^i) \in \{1, \ldots, n\}^3 \), where \( i_1^i, i_2^i, i_3^i \) provide indices to the associated vertices \( v_1^i, v_2^i, v_3^i \), and we denote the complete triangulation by \( T = (t_1, \ldots, t_m) \). Distances between shapes \( S_1 \) and \( S_2 \) are computed as difference between \( c_1 \) and \( c_2 \) after rigidly aligning \( S_1 \) and \( S_2 \) in \( \mathbb{R}^3 \).

3DMMs most often follow Dryden and Mardia [2002] for their definition of shape \( S \) as containing the geometric information remaining after having removed differences caused by translation, rotation, and sometimes uniform scaling. While scaling is typically not removed for human faces, this is often done in geometric morphometrics (e.g., [Dryden and Mardia 2002, Section 2]).

A shape space is traditionally defined as the set of all configurations of \( n \) vertices in \( \mathbb{R}^3 \) with fixed connectivity. Since we are interested in modeling human faces only in the context of 3DMMs, in the following, the term shape space refers to a \( d \)-dimensional parameter space (with \( d \ll n \)) that represents plausible 3D human faces. In this way, each 3D face has an associated parameter vector \( \mathbf{w} \in \mathbb{R}^d \).

In 3DMMs, statistical shape analysis is used as generative model, i.e. the shape space has an associated probability distribution called prior that is defined by a density function \( f(\mathbf{w}) \) and that measures the likelihood that a realistic 3D face would be represented by a particular vector \( \mathbf{w} \) in shape space. With a slight abuse of notation, we interpret \( c \) as a generator function in the following as

\[
\mathbf{c} : \mathbb{R}^d \rightarrow \mathbb{R}^{3n} \tag{1}
\]

that maps the low-dimensional parameter vector \( \mathbf{w} \) to the vector of all vertex coordinates \( \mathbf{c}(\mathbf{w}) \in \mathbb{R}^{3n} \). We again use \( v_i(\mathbf{w}) \in \mathbb{R}^3 \) to refer to the \( i \)-th vertex of the mesh given by \( \mathbf{w} \). While the resolution (number of vertices) of the model is usually fixed, a progressive mesh representation based on edge collapse simplification of the generator function has been considered [Patel and Smith 2011].

This part considers the case where all faces in the training data have a similar (typically neutral) expression; generator functions that additionally model varying expressions are discussed in Section 3.2. As in [Brunton et al. 2014b], our discussions distinguishes global models that model the entire face or head area from local models that perform statistics over localized areas.

3.1.1 Global models. Let \( \{S_i\}_i \) denote the training shapes and \( \{c_i\}_i \) their associated coordinate vectors. The seminal work on 3DMMs [Blanz and Vetter 1999] proposed a global shape model that uses principal component analysis (PCA) to compute the linear generator function as

\[
\mathbf{c}(\mathbf{w}) = \bar{\mathbf{c}} + \mathbf{E}\mathbf{w} \tag{2}
\]

where \( \bar{\mathbf{c}} \) is the mean computed over the training data, \( \mathbf{E} \in \mathbb{R}^{3n \times d} \) is a matrix that contains the \( d \) most dominant eigenvectors of the covariance matrix computed over the shape differences \( \{c_i - \bar{c}_i\} \), and \( \mathbf{w} \) is the low-dimensional shape parameter vector. One hypothesis of this model is that training faces can be linearly interpolated to generate new 3D faces. Another hypothesis is that the 3D faces in the reduced parameter space \( \mathbb{R}^d \) follow a multivariate normal distribution, which can be directly deduced from the eigenvalues corresponding to \( \mathbf{E} \). This implies that the density function \( f(\mathbf{w}) \) evaluating the likelihood of the parametric representation \( \mathbf{w} \) in shape space is simply the Mahalanobis distance of \( \mathbf{w} \) to the origin.

The 3DMM was originally computed over 200 subjects and has proven to be useful in a variety of applications thanks to its power to generate plausible shapes, and its simple underlying model.
recent study rebuilds such a model from a very large dataset containing 9,663 3D scans and revisits best practices [Booth et al. 2016], demonstrating that the originally proposed generator function for shape remains highly relevant in the research community.

One observation by Blanz and Vetter [1999] is that moving the representation vector \( \mathbf{w} \) away from the mean face increases their distinctiveness, eventually leading to caricatures of the identity. In order to model distinctive facial identities, Patel and Smith [2016] propose an alternative density function \( f(\mathbf{w}) \) based on the following observation. Consider the squared Mahalanobis distances from the mean for a set of \( d \)-dimensional vectors that follow a multivariate Gaussian distribution. These distances form a \( \chi^2_d \)-distribution, which has expected value \( d \). Hence, to preserve the shape distinctiveness related to identity, Patel and Smith restrict the representation \( \mathbf{w} \) to have Mahalanobis distance \( \sqrt{d} \) from the mean. Lewis et al. [2014b] propose a similar argument showing that, even if faces are truly Gaussian distributed (which has been shown for the Basel data by a Kolmogorov Smirnoff test for shape and per-vertex color, where the marginal distribution for the shape is close to a Gaussian [Egger et al. 2016b]), methods that make the assumption that typical faces lie near the mean are not valid.

Recently, Lüthi et al. [2018] proposed a nonlinear shape space for models deformations from the mean as Gaussian processes.

### 3.1.2 Local models

Using a global generator function in Equation (1) is known to lead to representations that do not model fine-scale geometric details. To improve the modeling of important localized areas, such as the eye or nose regions, Blanz and Vetter [1999] initially experimented manually segmenting the face into regions and learning separate PCA models per region. Their results demonstrate that this localized modeling allows for reconstructions of higher fidelity. This idea has been extended since with representations that achieve much higher accuracy than the global PCA model, and this comes in general at the cost of a less compact representation \( \mathbf{w} \).

First local models segmented the face manually [Basso and Verri 2007; Kakadiaris et al. 2007; ter Haar and Veltkamp 2008]. Smet and Gool [2010] and Tena et al. [2011] propose automatic ways of segmenting the faces into areas based on information learned over the displacements of corresponding vertices in the training set. Brunton et al. [2011] propose a model that combines shape variations that are localized in different areas with a multi-resolution framework that uses a wavelet decomposition of the 3D face models. Fine-scale geometric detail can alternatively be modeled using hierarchical pyramids that consider differences between a smooth face and increasingly high-resolution geometry representing e.g., wrinkles [Golovinskiy et al. 2006].

It is also possible to perform localized analysis using different statistical approaches than PCA. Neumann et al. [2013] propose the use of sparse PCA combined with a group sparsity constraint to identify localized deformation components over the training data. Ferrari et al. [2015] follow a related idea and learn a dictionary of deformation components oversampled regions for the application of face recognition. Wu et al. [2016b] combine a local deformation subspace model with an anatomical bone structure that acts as a regularizer of the deformation. The local deformation subspace is computed over overlapping localized patches, and the statistical model explicitly factors the rigid and non-rigid deformations applied to each patch.

### 3.2 Expression models

As simple linear models similar to the ones described can be used to model expression variation for one subject, this section considers models that capture variations of both identity and expression. Unlike simple linear models learned over a dataset of varying identities and expressions (e.g., [Booth et al. 2017]), our focus is on models that explicitly decouple the influence of identity and expression by modeling them in separate coefficients. We classify these methods into additive, multiplicative, and nonlinear models, depending on how the two sets of coefficients are combined.

#### 3.2.1 Additive models

Given two shapes of the same subject, one with expression \( \mathbf{e}_{\text{exp}} \) and one neutral shape \( \mathbf{e}_{\text{neu}} \), Blanz and Vetter [1999] transferred expressions between subjects by adding the expression offsets \( \Delta e := \mathbf{e}_{\text{exp}} - \mathbf{e}_{\text{neu}} \) to the neutral shape of another subject.

Several other methods then built on this idea, and model expression variations as an additive offset to an identity model with a neutral expression. Formally, additive models are given by

\[
\mathbf{c}(\mathbf{w}^s, \mathbf{w}^e) = \bar{c} + E^s \mathbf{w}^s + E^e \mathbf{w}^e,
\]

where \( \bar{c} \) is a mean, \( E^s \) and \( E^e \) are the matrices of basis vectors of the shape and expression space, and \( \mathbf{w}^s \) and \( \mathbf{w}^e \) are the shape and expression coefficients. Note that the basis vectors of the expression space can be interpreted as a data-driven blendshape model, where the basis vectors are orthogonal and do not carry interpretable semantic meaning in general [Lewis et al. 2014a].

Starting with Blanz et al. [2003], several methods propose to learn two PCA models, one over shape and one over expression to derive \( E^s \) and \( E^e \), and to compute \( \bar{c} \) as the mean over training data, either in neutral expression, or as sum of two means (one over shape and one over expression). Blanz et al. [2003] learned the expression space from a single subject captured in multiple expressions. Amberg et al. [2008] extended this work to include expression data from multiple subjects. This leads to a statistical expression model which does not enable control over specific facial expressions. It is therefore feasible for analysis-by-synthesis tasks but limited for controlling or synthesizing specific interpretable expression variation. Thies et al. [2015] use blendshapes as the basis vectors of the expression space. These expression blendshapes are not orthogonal and hence information of different blendshapes are potentially redundant.

#### 3.2.2 Multiplicative models

Another body of work model shape and expression variations in a multiplicative manner. Li et al. [2010] propose a method to adapt a pre-defined blendshape model to a specific subject given a small number of static face scans in different expressions, which provides a personalized facial rig. Bouaziz et al. [2013] combine a morphable shape model \( \mathbf{e}(\mathbf{w}) \) (Eq. 2) with a set of \( d_e \) linear expression transfer operators \( T_j : \mathbb{R}^{3n} \rightarrow \mathbb{R}^{3n} \) that transform the neutral shape to generate personalized blendshapes.
Formally, this model is defined as

\[ c(w^e, w^c) = \sum_{j=1}^{d_s} w_j^e T_j (c(w^e) + \delta^e) + \delta^c, \]

where \( \delta^e \) and \( \delta^c \) are corrective vectors to adapt the blendshapes to the tracked subject, and \( w_j^e \) is the \( j \)-th coefficient of \( w^e \).

A commonly used multiplicative model is the multiplicative model that extends the idea of PCA of performing a singular value decomposition to tensor data by performing a higher-order tensor decomposition (HOSVD) of 3D face data stacked into a training tensor. In particular, given a training set of different identities all captured in the same set of expressions, the vertex coordinates are stacked into a data tensor on which HOSVD is performed. This allows to model correlations of shape changes caused by identities and expressions. This model was first applied to 3D face modeling by Vlasic et al. [2005a], and can be defined as

\[ c(w_2, w_3) = M \times_2 w^e \times_3 w^c, \]

where \( M \in \mathbb{R}^{3 \times d_s \times d_e} \) denotes the multiplicative model tensor, and \( \times_i \) denotes the tensor mode-product. Thanks to its expressiveness and simplicity, this model is being used extensively for various applications [Bolkart and Wührer 2015a; Dale et al. 2011; Fried et al. 2016; Mpiperis et al. 2008; Yang et al. 2012]. To allow modeling localized variations, the multiplicative model has been applied to wavelet coefficients at different levels of detail [Brunton et al. 2014a].

Computing a multiplicative model with HOSVD requires a complete tensor of data, where each identity needs to be present in all expressions, and the data need to be in semantic correspondence specified by expression labels. This severely limits the kind of data that can be used for training. Recently, a number of methods have been proposed to address this limitation using an optimization approach [Bolkart and Wührer 2016], a custom tensor decomposition method [Wang et al. 2017], and an autoencoder structure [Fernández Abrevaya et al. 2018], respectively.

### 3.3.3 Nonlinear models

Facial shape and expression are mostly modeled with a linear subspace, often assuming a Gaussian prior distribution. Few methods exist to model facial variations with nonlinear transformations. Li et al. [2017] introduce FLAME, an articulated expressive head model that provides nonlinear control over facial expressions by combining jaw articulation with linear expression blendshapes. Ichim et al. [2017] use a muscle activation model driven by physical simulation. Koppen et al. [2018] instead of a single Gaussian distribution use a Gaussian mixture model to represent facial shape and texture. In another line of works, Shin et al. [2014] capture facial wrinkles in multi-scale maps and nonlinearly transfer them to other faces to enhance realism.

Recently, several deep learning-based models were published that fall into this group of nonlinear models [Bagautdinov et al. 2018; Lombardi et al. 2018; Ranjan et al. 2018; Tewari et al. 2019, 2018; Tran and Liu 2018a]. Section 6 covers these models in more detail.

### 3.3 Appearance models

This section describes approaches for modeling the facial appearance, where we distinguish between linear and nonlinear models. The appearance of a face is influenced by its albedo and illumination. However, most 3DMMs do not completely separate these factors, so that often times the illumination is baked into the albedo. Hence, in the following, we call the problem of statistically capturing this information appearance modeling. The most common way to build an appearance model is by performing statistics on appearance information of the training shapes, where the appearance information is usually either represented in terms of per-vertex values or as a texture in \( uv \)-space.

#### 3.3.1 Linear per-vertex models

Usually, color information is modeled as a low-dimensional subspace that is defined for the entire head. Huber et al. [2017] also use a linear per-vertex appearance model from images based on back-projection, where it is assumed that the 3D geometry of the face in the image is known. Their appearance model is not built using the color images directly but rather features computed from the images, for example, SIFT. This brings advantages that the features may be somewhat invariant to illumination changes and also that they depend on a local neighborhood which may widen the basin of convergence. In a similar vein, Wang et al. [2009] construct a linear model of spherical harmonic bases (see Section 4). This jointly models textures (more precisely diffuse albedo) and fine-scale shape (surface normal orientation) such that appearance under any illumination can be synthesized as a linear function of the basis.

#### 3.3.2 Linear texture-space models

A downside of per-vertex models is that they require compatible resolutions between the shape and appearance representation. This is rather uncommon in computer graphics, where usually a low(er) resolution geometry model (often times including normals) is used in conjunction with a high(er) resolution 2D texture map. Working with a 2D texture also has other advantages, such as the possibility of using image processing techniques to modify the texture maps. With that, such a representation is also amenable for being processed by convolutional neural networks (CNNs), as will be addressed in the next section.

We now turn our attention towards works that build linear appearance models in texture space. The original work by Blanz and Vetter [1999] used a texture-based representation by representing the face in a cylindrical way. Later, texture-based representations were used to add textural details like wrinkles [Pascal 2010], or to segment skin and detect moles [Pierrard 2008]. Cosker et al. [2011] model appearance variation in \( uv \)-space based on sequences of facial images recorded from different views. The images of the dynamic sequences are aligned based on a non-rigid registration so that the color variation can be modeled using a linear subspace model based on PCA. Dai et al. [2017] also use a \( uv \)-space appearance representation that is defined for the entire head. Huber et al. [2016] use a per-vertex appearance variation model based on PCA, but in addition, also define a common \( uv \)-mapping so that the model can be textured based on given facial images. Moschoglu et al. [2018] formulate a robust matrix factorization problem in order to learn...
attributed facial uv-maps from a collection of training textures. A study on the effect of different uv-space embeddings of the texture was presented by Booth and Zafeiriou [2014].

3.3.3 Nonlinear models. Traditionally, the facial appearance is modeled as a linear subspace, where oftentimes a Gaussian distribution is assumed. However, as empirically shown by Egger et al. [2016a], the Gaussian assumption is not very accurate and may lead to a sub-optimal facial appearance model. Hence, the authors proposed to replace a PCA-based appearance model with a Copula Component Analysis model [Han and Liu 2012]. Subsequently, this idea was extended to jointly model facial shape, texture, and attributes [Egger et al. 2016b]. Recent work learned a joint shape and texture model using neural networks with an adversarial loss [Gecer et al. 2019]. Alotaibi and Smith [2017] use the observation that skin color forms a nonlinear manifold in RGB space, approximately spanned by the colors of the pigments melanin and hemoglobin. They inverse render maps of these parameters and then construct a linear statistical model in the parameter space. The resulting biophysical 3DMM is guaranteed to produce plausible skin colors. In addition to global facial appearance models, there are also approaches that consider models of local skin variations. For example, Dessein et al. [2015] use a texture model based on small overlapping patches that are extracted from a face database, and Schneider et al. [2018] have presented a stochastic model that is able to synthesize freckles.

More recently, a range of appearance modeling approaches based on deep learning have been proposed, where many of these methods are also built within an analysis-by-synthesis framework. These aspects will be discussed in-depth in Secs. 5 and 6.

3.4 Joint shape and appearance models
Blanz and Vetter [1999] originally proposed building separate, independent models for shape and texture. Interestingly, in 2D the Active Appearance Model [Cootes et al. 1998] was originally proposed with a combined shape and appearance model. The advantage of such a joint model is that correlations between shape and texture can be learned and exploited as a constraint during fitting with fewer parameters. On the other hand, separate models are more flexible and, since shape and texture parameters can be adjusted independently, sequential algorithms can fit the two models independently. However, 3DMMs that jointly model shape and texture have subsequently been considered. Schumacher and Blanz [2015] use canonical correlation analysis to study shape/texture correlations and also correlations between face parts. Egger et al. [2016a] use copula component analysis that can deal with the different scales of shape and texture data. Zhou et al. [2019] propose a deep convolutional colored mesh autoencoder that learns a joint nonlinear model of shape and texture.

3.5 Correspondence
The previously discussed models typically require the data with point-to-point correspondence between all shapes. We refer to the process of establishing such a dense correspondence between scans as registration in the following.

Many methods exist to establish point-to-point correspondence for general classes of objects (e.g., [Tam et al. 2013; van Kaick et al. 2011]), yet the space of face deformations is strongly constrained. Most commonly used face registration methods follow the principle of deforming a template mesh to each scan in the dataset. This registration process typically starts with a rough alignment (often using sparse correspondences) and leads to dense correspondences in the end.

While several image-based methods can also be seen as jointly learning correspondence (between images) and building a statistical model (e.g., [Tewari et al. 2019; Tran and Liu 2018a]), we cover such deep learning-based methods in Section 6 in more details.

3.5.1 Sparse correspondence computation. Several methods exist to establish a sparse correspondence for a dataset of 3D scans by predicting landmarks, i.e. a common set of salient points, for each scan. This sparse correspondence then typically serves as automatic initialization for dense correspondence methods.

Most of the methods use some local descriptors, or combination of local descriptors and connectivity information between descriptors, to predict salient points. While landmark localization in images is widely researched (e.g., Bulat and Tzimiropoulos [2017]), our focus is on methods that establish sparse correspondence between 3D scans.

Existing methods use combinations of different geometric descriptors. Passalis et al. [2011] use shape index and spin image features, Berrett et al. [2011] use curvature and scale-invariant feature transform (SIFT) features, and Creusot et al. [2013] consider combinations of local features such as Gaussian curvature, mean curvature, and a volumetric descriptor, and learn the statistical distribution of these descriptors for each landmark.

Further, existing methods use geometric relations or relations between landmarks along with geometric feature descriptors. Guo et al. [2013] project a scan into an image and predict landmarks with a 2D PCA model and geometric relations with additional texture constraints. Salazar et al. [2014], similarly to Creusot et al. [2013], learn the statistical distribution of local surface descriptors with additional Markov network to additionally consider connections between landmarks. Bolkart and Wurth [2015a] extend this further to sequences by additionally considering temporal edges within the Markov network.

3.5.2 Dense correspondence computation. Methods that deform a template to establish correspondence mostly differ in the parameterization of the deformation. We group existing methods according to the type of scan data they register. We distinguish here between static methods, i.e. methods that aim at registering static 3D scan, and dynamic methods that register 3D motion sequences. Blanz and Vetter [1999] use a bootstrapping approach to iteratively fit a 3DMM to a scan, refine the correspondence between the model fit and the scan with a flow field, and refine the model. Blanz et al. [2003] later extend this approach to expressive scans. Amberg et al. [2008] register expressive scans with a non-rigid ICP. Hutton et al. [2001] establish a thin-plate spline (TPS) mapping to warp each scan to a reference and establish correspondence using nearest neighbor search.

Passalis et al. [2011] register scans by deforming an annotated face model (AFM) [Kakadiaris et al. 2005], i.e. an average 3D face template that is segmented into different annotated areas, by solving...
a second-order different equation. Mpiperis et al. [2008] initially fit a subdivision surface to a scan, where the deformation of the base mesh (i.e. the mesh of the lowest subdivision level) is guided by a sparse landmark correspondence. After registering a training set, they parametrize the deformation of the base mesh with a PCA model over the training data. Salazar et al. [2014] use a generic expression blendshape model to fit the expression of the scan, followed by a non-rigid ICP to closely fit the surface of the scan. [Gerig et al. 2018] establish dense correspondence with a Gaussian process deformation model with the spatially varying kernel.

Several methods exist to sequentially register motion sequences. Weise et al. [2009] use an identity PCA model to register a neutral scan, and then track motion sequences by optimizing sparse and dense optical flow between consecutive frames. Fang et al. [2012] and Li et al. [2017] initialize the optimization by the registration of the previous frame to exploit temporal information. Fang et al. [2012] use an AFM, Li et al. [2017] a non-rigid ICP regularized by FLAME. Fernández Abrevaya et al. [2018] use a spatiotemporal method to register entire motion sequences by iteratively refining the registration of entire sequences by explicitly encoding temporal information with a Discrete Cosine Transform (DCT).

Further, non-template fitting based registration methods exist. Sun et al. [2010] use a conformal mapping to parameterize two meshes and establish dense correspondence between the resulting planar meshes by extrapolating sparse landmark correspondences. Ferrari et al. [2015] segment the face scans into non-overlapping parts divided by geodesic curves between selected landmark pairs, and consistently re-sample each part.

3.5.3 Jointly solving for correspondence and statistical model. Li et al. [2013] and Bouaziz et al. [2013] jointly update person-specific blendshape models and register motion sequences in a real-time facial animation framework. During tracking, Li et al. [2013] use an adaptive PCA model that combines the person-specific blendshapes with additional corrective basis vectors that are successively updated, and Bouaziz et al. [2013] optimizes for corrective deformation fields (Equation 4).

Bolkart and Wuhrer [2015b] and Zhang et al. [2016b] instead optimize correspondence for a dataset of different subjects in multiple expression in a groupwise fashion. Bolkart and Wuhrer [2015b] jointly update the point correspondence within the mesh surface by minimizing an objective function that measures the compactness of a multilinear face model. Zhang et al. [2016b] optimize functional maps across the entire dataset.

3.6 Synthesis of novel model instances
3DMMs can be used to synthesize new 3D faces that are different from any of the observed training data, yet realistic. This can be achieved by altering the coefficients in parameter space (i.e. shape space, expression space or appearance). Common operations in parameter space include interpolating or extrapolating between the coefficients of training samples. Furthermore, any of the generative models presented in this section can be used to directly synthesize new 3D faces by drawing random samples in parameter space according to the prior distribution. Depending on the model, this sampling allows to synthesize or alter identity, expression, or appearance of a static 3D face. Synthesis works are heavily used for entertainment purposes, and these works are discussed in Section 7.2.

Synthesis of static 3D faces notably includes the generation of face caricatures by moving the identity coefficient linearly away from the mean [Blanz and Vetter 1999] which is mainly explored to study the human face processing system as discussed in Section 7.5. With 3DMMs that encode and decouple identity and expression information, it is easy to synthesize dynamic sequences by fixing the identity coefficients while modifying the expression coefficients. Some works aim to synthesize coherent dynamic 3D face videos of a fixed identity with the help of 3DMMs. These include works that synthesize 4D videos from a static 3D mesh paired with semantic label information [Bolkart and Wuhrer 2015a], and from a static 3D mesh and audio information [Cudeiro et al. 2019].

3.7 Publicly available models
In Table 2 we list publicly available shape and/or appearance models of human faces. Figure 4 visualizes geometry or appearance variations of some models. We also refer to the curated list of 3DMM software and data that we collected, share and update [Community 2019].

3.8 Open challenges
While 3D face modeling has received considerable attention during the past two decades, some challenges remain. First, the statistics of most models are limited to the face and do not include information on eyes, mouth interior or hair. These details are however crucial for many applications, and it is not straightforward to combine a 3DMM with specific models e.g., for hair. Second, the interpretability of the representations would benefit from being improved. PCA is the most commonly used method to perform statistics on 3D faces, and as it is an unsupervised method, the principal components do not coincide with attributes that humans would use to describe a face. Third, methods that incorporate different levels of detail typically come at the cost of a less compact representation, and it is unknown how many parameters are required to accurately represent facial geometry and appearance at varying levels of detail. Fourth, the different models presented in this section have different advantages and drawbacks, making them most suitable for specific applications. It is unknown whether one integrated optimal model for all applications exists. Fifth, all currently available models, even the large scale ones have a very strong racial bias towards white. This can be alleviated in the future by scanning efforts in different parts of the world. Another potential solution to overcome a racial bias can be the generative model itself, as these allow to generate and add synthetic data to biased datasets. Sixth, learning from inhomogeneous data presents another open challenge. There are many available datasets with different resolution, coverage, noise characteristics, biases and so on (see Table 1). To make the best use of this data requires methods that can learn models from all data sources simultaneously but this requires explicit ways to deal with data inhomogeneity. Some very recent work begins to look at this problem [Liu et al. 2019b]. Finally, there are some fundamental open questions related to the statistical modeling of shape. Two face shapes differ by nonlinear shape deformation superposed on
Fig. 4. Model variations of existing face models. Top left: CFHM [Ploumpis et al. 2019] shape variations. Top right: FaceWarehouse [Cao et al. 2014b] shape and expression variations (while the original model is not available to the best of our knowledge, the visualized multilinear face model is trained from the published FaceWarehouse dataset). Middle: BFM 2019 [Gerig et al. 2018] shape, expression, and appearance variations. Bottom: FLAME [Li et al. 2017] shape, expression, pose, and appearance variation. For shape, expression, and appearance variations, three principal components are visualized at ±3 standard deviations. The FLAME pose variations are visualized at ±π/6 (components three and four) and at 0, π/8 (component six).

4 IMAGE FORMATION

A 3DMM provides a parametric representation of face geometry and appearance. One key usage of such a model is synthesis, which involves two steps. First, generating a new model instance via sampling from the parameter space or manual interaction with model parameters (see Section 3.6). Second, rendering the generated model into a 2D image via a simulation of the image formation process, i.e. the computer graphics pipeline. The synthesis also forms an important part of 3DMM-based face analysis, either through classical analysis-by-synthesis (see Section 5) or as a model-based decoder.
<table>
<thead>
<tr>
<th>Model</th>
<th>Geometry</th>
<th>Appearance</th>
<th>Data</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basel Face Model (BFM) 2009</td>
<td>shape</td>
<td>per-vertex</td>
<td>200 individuals, each in neutral expression</td>
<td>includes separate models for facial parts</td>
</tr>
<tr>
<td>FaceWarehouse [Cao et al. 2014b]</td>
<td>shape, expression</td>
<td>-</td>
<td>150 individuals, each with 20 expressions</td>
<td></td>
</tr>
<tr>
<td>Global and local linear model [Brunton et al. 2014b]</td>
<td>shape</td>
<td>-</td>
<td>100 individuals</td>
<td></td>
</tr>
<tr>
<td>Multilinear Wavelet model [Brunton et al. 2014a]</td>
<td>shape, expression</td>
<td>-</td>
<td>99 individuals, 25 expressions</td>
<td></td>
</tr>
<tr>
<td>Multilinear face model [Bolkart and Wuhrer 2015b]</td>
<td>shape, expression</td>
<td>-</td>
<td>2500 scans (100 individuals, 25 expressions)</td>
<td></td>
</tr>
<tr>
<td>Multilinear face model [Bolkart and Wuhrer 2016]</td>
<td>shape, expression</td>
<td>-</td>
<td>2510 scans (205 individuals, up to 23 expressions)</td>
<td></td>
</tr>
<tr>
<td>Large Scale Facial Model (LSFM) [Booth et al. 2016]</td>
<td>shape</td>
<td>-</td>
<td>9663 individuals</td>
<td></td>
</tr>
<tr>
<td>Surrey Face Model [Huber et al. 2016]</td>
<td>shape, expression</td>
<td>per-vertex</td>
<td>169 individuals</td>
<td>multi-resolution</td>
</tr>
<tr>
<td>Liverpool-York Head Model (LYHM) [Dai et al. 2017]</td>
<td>shape</td>
<td>per-vertex</td>
<td>1212 individuals</td>
<td>full head (no hair, no eyes)</td>
</tr>
<tr>
<td>Faces Learned with an Articulated Model and Expressions (FLAME) [Li et al. 2017]</td>
<td>shape, expression, head pose</td>
<td>texture</td>
<td>3800 individuals for shape, 8000 for head pose, 21000 frames for expression</td>
<td>female, male, gender neutral model, full head (no hair)</td>
</tr>
<tr>
<td>Basel Face Model (BFM) 2017</td>
<td>shape, expression</td>
<td>per-vertex</td>
<td>200 individuals for shape and appearance, a total of 160 expression scans</td>
<td>BFM 2019 with full head and multi-resolution</td>
</tr>
<tr>
<td>York Ear Model [Dai et al. 2018]</td>
<td>shape</td>
<td>-</td>
<td>20 3D ear scans, augmented with 605 landmark-annotated 2D ear images</td>
<td>ear only</td>
</tr>
<tr>
<td>Multilinear autoencoder [Fernández Abrevaya et al. 2018]</td>
<td>shape, expression</td>
<td>-</td>
<td>5000 scans from 195 individuals, 500000 after augmentation</td>
<td></td>
</tr>
<tr>
<td>Convolutional Mesh Autoencoder (CoMA) [Ranjan et al. 2018]</td>
<td>shape, expression</td>
<td>-</td>
<td>12 individuals, 12 extreme expressions, 20466 meshes in total</td>
<td>full head (no hair)</td>
</tr>
<tr>
<td>Combined Face &amp; Head Model (CFHM) [Ploumpis et al. 2019]</td>
<td>shape</td>
<td>-</td>
<td>Merged from LYHM and LSFM models</td>
<td>full head (no hair)</td>
</tr>
<tr>
<td>Morphable Face Albedo Model [Smith et al. 2020]</td>
<td>-</td>
<td>per-vertex diffuse and specular albedo</td>
<td>73 individuals (50 scanned + 23 3DRFE [Stratou et al. 2011])</td>
<td>extends BFM2017</td>
</tr>
</tbody>
</table>

Table 2. Overview of publicly available 3D shape and/or appearance models of human faces.

within a deep learning architecture (see Section 6). In this section, we focus on modeling the image formation process. This potentially encompasses the whole of the rendering literature, so we restrict our attention to techniques and models that have been applied in the context of 3DMMs. We cover the geometry and photometry of image formation in Sections 4.1 and 4.2, the rendering pipelines used for 3DMM fitting in Section 4.3 and finally in Section 4.4 we highlight where there are future opportunities for exploiting state-of-the-art rendering techniques to improve 3DMM synthesis.

4.1 Geometric image formation

A camera model describes the geometry of image formation, specifically, how positions in the 3D world project to 2D locations in the image plane. A variety of camera models have been used in the
3DMM literature which are described here in order of increasing accuracy with respect to a real camera. We denote the projection of a 3D point \( v = [u, v, w]^T \) onto the 2D point \( x = [x, y]^T \) by \( x = \text{project}(C, v) \in \mathbb{R}^2 \), where \text{project} represents one of the camera projection models below and \( C = (C_{\text{intrinsic}}, C_{\text{extrinsic}}) \) contains the camera parameters. \( C_{\text{extrinsic}} = (R, t) \) describes the pose in terms of a rotation \( R \in SO(3) \) and translation \( t \in \mathbb{R}^3 \) that transform from world to camera coordinates. \( C_{\text{intrinsic}} \) is a set of internal parameters specific to each projection model. The task of estimating \( C \) is known as camera calibration or camera resectioning and is usually done from known or estimated 2D-3D correspondences. Estimating \( C_{\text{extrinsic}} \) with known \( C_{\text{intrinsic}} \) is called pose estimation or, in the case of a perspective camera model, the perspective-n-point problem.

**Scaled orthographic.** The scaled orthographic projection model comprises an orthographic projection whose sole parameter is a uniform scaling \( s \in \mathbb{R}_{>0} \):

\[
\text{ortho}[v, R, t, s] = sP(Rv + t) = s \begin{bmatrix} r_1 & t_1 \\ r_2 & t_2 \end{bmatrix} \tilde{v} = C\tilde{v}, \quad P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}
\]

(7)

where \( \tilde{v} = [u, v, w, 1]^T \) is the homogeneous representation of \( v \) and \( r_1, r_2 \) are the first two rows of \( R \). This model is not physically meaningful but is linear in vertex position, translation and scale and avoids size/distance/perspective ambiguities introduced by more realistic camera models. Since \( R \) must be restricted to \( SO(3) \), the projection is nonlinear in any parameterization of \( R \). In the context of 3DMMs, scaled orthographic projection has been used for example by Bas et al. [2017b]; Blanz et al. [2004a]; Knöthe et al. [2006]; Patel and Smith [2009]. The scaled orthographic model can be interpreted as an approximation to perspective projection when the distance between the surface and camera is large relative to the depth variation. Concretely, when \( \max_{w} \omega - \min_{w} \omega \ll \tilde{w} \) with \( \tilde{w} = \text{mean}_{w} \omega \) the mean distance between the surface and the camera, then the nonlinear division in perspective projection can be approximated by a fixed scale \( s = f/\tilde{w} \) where \( f \) is the focal length of the camera. This gives physical meaning to the scaled orthographic model.

**Affine.** The affine camera generalizes the orthographic model by allowing arbitrary affine transformations. Specifically, this additionally allows non-uniform scaling and skew transformations which can approximate perspective effects whilst remaining linear. An affine camera can be represented by an arbitrary matrix \( C \in \mathbb{R}^{2 \times 4} \) with the projection given simply by \( x = \text{affine}[v, C] = C\tilde{v} \). The projection is linear in \( C \) and since its 8 entries are unconstrained, they can be estimated using linear least squares (though note that numerical stability entails first performing a normalization procedure). In the context of 3DMMs, the affine camera has been used for example by Aldrian and Smith [2013]; Huber et al. [2016].

**Perspective.** A nonlinear perspective projection is given by the pinhole camera model \( x = \text{pinhole}[v, K, R, t] \). The matrix:

\[
K = \begin{bmatrix} f_x & \gamma & c_x \\ f_y & \gamma & c_y \\ 0 & 0 & 1 \end{bmatrix}
\]

contains the intrinsic parameters of the camera, namely the focal length in the \( x \) and \( y \) directions \( f_x, f_y \), the skew \( \gamma \) and the principal point \( c_x, c_y \in \mathbb{R}^2 \). Common assumptions are that the pixels are square (in which case a single focal length \( f = f_x = f_y \) parameter is used), that the camera sensor is perpendicular to the camera view vector (in which case \( \gamma = 0 \)) and that the principal point is in the centre of the image \( (c_x = w/2 \) and \( c_y = h/2 \)). Note that \( f \) is actually a product of two quantities: the physical focal length in world units (e.g., mm) and the conversion factor from world units to pixels (i.e. with units of pixels/mm). The nonlinear perspective projection can be written in linear terms by using homogeneous representations:

\[
\lambda \tilde{x} = K[R \ t] \tilde{v} = C\tilde{v},
\]

(8)

where \( \tilde{x} = [x, y, 1]^T \), \( \lambda \) is an arbitrary scaling factor and \( C \in \mathbb{R}^{3 \times 4} \) is known as the camera matrix. The final image coordinate is obtained by the nonlinear homogenization of \( \tilde{x} \). Unlike the linear models, the pinhole model captures the effect of distance on projected shape. This becomes important when a face is close to the camera. At “selfie” distance (e.g., 0.5m), the difference between perspective and orthographic projection of 3D face landmarks is about 6% of the interocular distance [Bas and Smith 2019]. For this reason perspective projection is commonly used in the context of 3DMMs, for example, in the original Blanz and Vetter [1999] paper and more recently in a shape-from-landmarks setting [Cao et al. 2014a, 2013; Saito et al. 2016]. Unfortunately, since calibration information is rarely available, the increased complexity of this model introduces ambiguities between shape, scale and focal length that have only recently been studied [Bas and Smith 2019; Smith 2016], though the ambiguity has often been hinted at in the literature. For example, the original Blanz and Vetter [1999] paper relied on a fixed, manually provided subject-camera distance. Booth et al. [2018b] state “we found that it is beneficial to keep the focal length constant in most cases, due to its ambiguity with \( t_z \)”.

4.2 Photometric image formation

The appearance of a face is determined by the interaction of light with the material of the face, predominately skin. Hence, the photometry of both illumination and reflectance must be modeled in order to simulate the image formation process.

**Reflectance models.** The reflection of light from a surface is often described using a Bidirectional Reflectance Distribution Function (BRDF). This describes the directional dependence of local light reflection from an opaque surface. It is represented by a four dimensional function \( f_r(\omega_i, \omega_o) \) that gives the ratio of outgoing reflected radiance in direction \( \omega_o \) to incoming incident radiance from direction \( \omega_i \). A BRDF allows us to express irradiance \( E_o(\omega_o) \) in direction
\( \omega_o \) as a function of light reflected from all incident directions:

\[
L_o(\omega_o) = \int_{\Omega(n)} f_r(\omega_1, \omega_o)L_i(\omega_1) \cos \theta_1 d\omega_1,
\]

where \( L_i(\omega_1) \) is incident irradiance from direction \( \omega_1 \), \( \Omega(n) \) is the hemisphere around the local surface normal \( n \) and \( \theta_1 \) is the angle between \( \omega_1 \) and \( n \). Note \( \cos \theta_1 = n \cdot \omega_1 \), where \( \cdot \) denotes the inner product. Physically-valid BRDFs must exhibit a number of properties: nonnegativity \( (f_r(\omega_1, \omega_o) \geq 0) \), Helmholtz reciprocity \( (f_r(\omega_1, \omega_o) = f_r(\omega_o, \omega_1)) \) and conservation of energy:

\[
\forall \omega_1, \int_{\Omega(n)} f_r(\omega_1, \omega_o) \cos \theta_1 d\omega_o \leq 1.
\]

A particularly simple and commonly used physically valid BRDF is the Lambertian model for a perfectly diffuse reflector. The Lambertian model assumes incident light is scattered equally in all directions resulting in a constant BRDF: \( f_{\text{Lambert}}(\omega_1, \omega_o) = \rho_d / \pi \). \( \rho_d \in [0, 1] \) is the diffuse reflectivity or albedo, that is usually wavelength-dependent and can be thought of as the color of the object. Work predating the original 3DMM used a linear statistical 3D face model in the context of 3DMMs. Aldrian and Smith [2013] also used the same model for specular reflection, showing that the statistical texture model is the correct model for albedo in the Lambertian case, the contribution from the reflectance function \( \rho_i \) and \( \rho_s \) are the orthonormal basis functions, and \( \rho_d \) is a unit vector in the viewer direction. Usually, the light source intensity and albedos would be RGB values. Often \( \rho_d = \rho_d \), representing the intrinsic color of the surface. In a 3DMM, this is described by the statistical texture model \( f_r(\omega_1, \omega_o) \). Note that these simple models are purely local, this means that they neglect self occlusion of the light source, i.e. cast shadows. These can be added at the cost of computing the occlusion function which is not differentiable.

Blanz and Vetter [1999] originally used the Phong model which augments the Lambertian term with a constant ambient term and a phenomenological specular model enabling the simulation of glossy reflectance. The Phong model can be described in terms of the following BRDF:

\[
f_{\text{Phong}}(\omega_1, \omega_o) = \frac{\rho_d + \rho_s (r \cdot \omega_o)^\eta}{n \cdot \omega_1} + \rho_d
\]

where \( r \) is the reflection of \( \omega_1 \) about \( n \), \( \eta \) is the shininess that controls the width of the specular lobe and \( \rho_d, \rho_s \) are ambient and specular "albedos". In the context of 3DMMs, usually only \( \rho_d \) is allowed to vary spatially. Note that the Phong BRDF does not satisfy the constraints above for physical validity. In graphics, extremely complex, physically-valid BRDF models have been developed specifically for materials of relevance to face 3DMMs, for example for skin [Krishnaswamy and Baranoski 2004] and hair [Marschner et al. 2003]. Note that skin is a layered, partially translucent material and so a local BRDF model is inadequate to describe the actual subsurface scattering effects that take place. More complex 8-dimensional bidirectional subsurface scattering reflectance distribution functions (BSSRDF) have been proposed for such materials. However, both these and the more complex BRDFs have proven to be too complex to integrate into 3DMM fitting pipelines and so the majority of work has used Lambertian or non-physical models of moderate complexity.

**Lighting.** In (9), \( L_i(\omega_1) \) represents the hemispherical incident illumination environment at the surface point. Natural illumination is usually complex, comprising multiple, possibly extended sources as well as secondary illumination reflected from other surfaces. A common assumption is that the illumination environment is distant relative to the size of the object in which case it can be represented by a constant 2D environment map, a discrete approximation of \( L_i(\omega_1) \) that is used for every point on the surface. However, the space of possible natural illumination is very high dimensional and rendering with an environment map is computationally expensive, so a number of further simplifications are commonly used.

The simplest illumination model is a point source, in which \( L_i(\omega_1) \) is a delta function characterized by a unit vector in the light source direction, \( s \), and an intensity, \( L_i \). Ignoring constants and assuming image intensity is proportional to surface radiance, we can plug in the simple BRDF models above and obtain the following shading models:

\[
l_{\text{Lambert}} = L_i \rho_d n \cdot s, \quad l_{\text{Phong}} = L_i \left[ \rho_d + \rho_s (r \cdot v)^\eta \right] \cdot v
\]

where \( v \) is a unit vector in the viewer direction. Usually, the light source intensity and albedos would be RGB values. Often \( \rho_d = \rho_d \), representing the intrinsic color of the surface. In a 3DMM, this is described by the statistical texture model \( f_r(\omega_1, \omega_o) \). Note that these simple models are purely local, this means that they neglect self occlusion of the light source, i.e. cast shadows. These can be added at the cost of computing the occlusion function which is not differentiable.

A better approximation of complex natural illumination is provided by the spherical harmonic illumination model [Ramamoorthi and Hanrahan 2001]. Spherical harmonics provide an orthonormal basis for functions on the sphere, analogous to a Fourier basis in Euclidean space:

\[
l_{\text{SH}}(n) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} l_{l,m} B_{l,m}(n),
\]

where \( B_{l,m}(n) \) are the orthonormal basis functions, and \( l_{l,m} \) are coefficients describing reflectance and illumination. The subscript \( l \) denotes the degree and \( m \) the order of the spherical harmonics. In the Lambertian case, the contribution from the reflectance function is constant and 98% of the energy of the reflectance function can be captured for any illumination environment using an order 2 \( (l = \{0, 1, 2\}) \) approximation. In practice, this means that a good approximation for appearance can be obtained using 9 illumination coefficients per color channel. Combining this model with the linear texture model for albedo \( d(w^t) \) yields the following:

\[
l_{\text{SH}} = d(w^t) \odot \text{vec}(B(w^t)L),
\]

where \( \odot \) is the Hadamard (element-wise) product. The matrix \( B(w^t) \in \mathbb{R}^{m \times 9} \) contains the spherical harmonic basis for each vertex which depends on the vertex normal direction and hence the geometry which in turn is determined by the shape parameter vector. The matrix \( L \in \mathbb{R}^{9 \times 3} \) contains the lighting coefficients for each color channel. Zhang and Samaras [2006] were the first to fit this model in the context of 3DMMs. Aldrian and Smith [2013] additionally used the same model for specular reflection, showing that the coarse structure of the illumination environment can be recovered.
from a face image. They also introduced priors to help resolve lighting/texture ambiguities. Egger et al. [2018] went further by learning a low dimensional illumination prior from spherical harmonic lighting coefficients estimated from real-in-the-wild images.

An alternate model that takes a step towards capturing global illumination effects is the ambient occlusion model. Here, it is assumed that \( L_i(\omega_i) \) is constant everywhere, i.e. that illumination is perfectly diffuse. In this case, shading depends only upon the degree to which the incident hemisphere is occluded. The ambient occlusion, \( A_v \), at vertex \( v \) is given by:

\[
A_v = \frac{1}{\pi} \int_{\Omega(n)} V(v, \omega)(n \cdot \omega) d\omega,
\]

where \( V(v, \omega) \) is the visibility function defined as zero if vertex \( v \) is occluded in direction \( \omega \), and one otherwise. One can also define the bent normal as the average unoccluded direction. Using the bent normal with the spherical harmonic model and scaling the result by the ambient occlusion provides a rough approximation of global illumination effect. Ambient occlusion and bent normal direction depends on the geometry and hence the 3DMM shape parameters. Aldrian and Smith [2012] proposed to learn a linear model of ambient occlusion and bent normals and included this in their 3DMM synthesis model. Zivanov et al. [2013] similarly construct a joint linear model of spherical harmonic bases and ambient occlusion.

The most complex global model of appearance considered in the context of 3DMMs is the precomputed radiance transfer (PRT) model [Sloan et al. 2002]. This uses an efficient representation (such as spherical harmonics) to approximate the local light transport at each vertex, accounting for shadowing and inter-reflection. These are precomputed but can then be used with any incident illumination at render time. Schneider et al. [2017] learn a linear model of PRT transfer matrices as a function of the 3DMM shape coefficients and use this in a rendering framework.

We denote by \( L \) the set of illumination parameters for any of the above illumination models.

**Color transformation.** In a real camera, the actual image irradiance measured by the sensor is usually transformed in a complex way in order to achieve a pleasing visual appearance. Often, this amounts to multiplication by a \( 3 \times 3 \) color transformation matrix followed by a nonlinearity. The color transformation matrix can be decomposed into a product of three \( 3 \times 3 \) matrices: \( T = T_{xyz2rgb} T_{raw2xyz} T_{rgb} \), where \( T_{rgb} \) is a diagonal matrix that performs white balancing (compensating for the color of the illumination). \( T_{raw2xyz} \) is specific to each camera and maps from the native color space to the standardized XYZ space and \( T_{xyz2rgb} \) is a fixed matrix that transforms to sRGB space. Unfortunately, introducing such a color transformation into the 3DMM image formation model further exacerbates the lighting/albedo ambiguity by providing an additional explanation for observed color. Finally, a nonlinearity is applied, which can be approximated by \( \gamma \text{ln} \rightarrow \gamma \text{ln}\), where usually \( \gamma = 2.2 \). This non-linear transformation is important because it means the, often linear, reflectance and illumination models described above cannot explain the final image appearance.

Despite their importance, camera color transformations and nonlinear gamma are almost always ignored in the context of 3DMMs. There are some notable exceptions. Schneider et al. [2017] apply gamma correction to input images to transform back to a linear space. Blanz and Vetter [2003] estimate a per-channel scale and offset as well as scalar color contrast, allowing them to synthesize grayscale images. The same model was used by Aldrian and Smith [2013] and Hu et al. [2013].

4.3 Rendering and visibility

3DMM fitting algorithms differ in whether they synthesize a discrete image in image space (i.e. one color per-pixel) or perform rendering in object space (i.e. one color per-vertex or per-triangle). The former holds the advantage that it is straightforward to incorporate a texture model built in a high-resolution UV space and also that the output is a regular pixel grid that can be passed to CNN, for example for an adversarial loss [Shamai et al. 2020]. Methods that work in object space compute an appearance error by projecting the model vertices into the image and sampling image intensities onto the visible vertices. Visibility can also be computed in either object space or image space, with the latter usually being more efficient.

The original Blanz and Vetter [1999] paper used object space rendering in which a single color was computed for each triangle center (equivalent to flat shading) with image space z-buffering used for visibility testing. Many subsequent methods also worked in object space but usually with per-vertex colors computed using the reflectance models described above with per-vertex surface normals. This has begun to change recently when more conventional rasterization pipelines have been included in 3DMM synthesis. Rasterization associates with each pixel \( (x, y) \in I \), where \( I = \{1, \ldots, w\} \times \{1, \ldots, h\} \), a triangle index or a NULL value if the pixel is not covered by a triangle:

\[
\text{raster}_{C, T, \omega^{s}, \omega^{r}} : I \mapsto \{1, \ldots, m, \text{NULL} \}.
\]

recalling that \( \omega^{s} \), \( \omega^{r} \) are the shape and expression parameters respectively and \( T \) the mesh triangulation. Since this is a discrete function it is not smooth and not differentiable. In addition, for each pixel, three weights are calculated that are associated with the vertices of the rasterized triangle: \( a_{C, T, \omega^{s}, \omega^{r}}(x, y) \in \mathbb{R}_{\geq 0} \). These weights depend on the projected positions of the vertices

\[
V_{\mu, i} = \text{raster}_{C, T, \omega^{s}, \omega^{r}}(x, y), \quad i \in \{1, 2, 3\}.
\]

Often, these weights are barycentric coordinates of the pixel center within the triangle. These weights are a smooth function of the vertex positions and hence of the shape and camera parameters. Hence, rendering is differentiable up to a change in rasterization, i.e. so long as the triangle index associated with each pixel does not change. Tran and Liu [2018a] incorporate such a conventional rasterization pipeline into an in-network differentiable renderer.

Collecting together all of the parameters relating to the camera, illumination, face geometry and texture, \( \Theta = (C, L, \omega^{s}, \omega^{r}, w^{s}, w^{r}) \), we can write the rendered appearance in object space of vertex \( j \) as \( \mathbf{i}_{\text{model}}^{j}(\Theta) \). For an image space rendering we denote the appearance of the model at pixel \( (x, y) \) by \( \mathbf{i}_{\text{model}}^{j}(\Theta) \). In the simplest case, the image space rendering is computed directly from the object space
rendering using Gouraud interpolation shading:

\[ I_{\text{model}}^{x,y}(\Theta) = a_{C,T,w^s,w^r}(x,y)^T \begin{bmatrix} I_{\text{model}}^{f_1}(\Theta) \\ I_{\text{model}}^{f_2}(\Theta) \end{bmatrix} \cdot \]

where \( j = \text{raster}_{C,T,w^s,w^r}(x,y) \). Other rasterization strategies may be more complex. For example, Genova et al. [2018] use rasterization in a differentiable deferred shading renderer more akin to Phong interpolation shading. Here, vertex normals and colors are rasterized and interpolated, then reflectance calculations are done in image space.

Note that overcoming the non-differentiable nature of rasterization is an open problem. Hiroharu Kato and Harada [2018] present an approximately differentiable renderer based on rasterization. Liu et al. [2019a] propose a rasterizer in which triangles make a soft (and hence differentiable) contribution to image appearance. More ambitiously, differentiable rendering using other pipelines is now also being considered, for example, differentiable path tracing [Li et al. 2018].

Very recently, the explicit fixed models used in conventional rendering are being augmented or replaced by learning components, so-called neural rendering. For example, Kim et al. [2018a] train an image to image network that transforms a low-quality 3DMM rendering into a photorealistic video frame.

4.4 Open challenges

The image formation models used in the context of 3DMMs are much simpler than those used in graphics and many other areas of computer vision. For example, we are not aware of any work that allows for a center of projection different to the center of the image, even though many face image datasets consist of images cropped (probably non-centrally) from larger images. Similarly, nonlinear distortion is always ignored. The effect of this assumption is not understood. In other fields like structure-from-motion, it is standard to impose constraints derived from metadata, knowledge of physical camera parameters and so on. This is not currently being done to a significant extent with 3DMMs.

Advances in rendering in computer graphics are slowly propagated into the world of 3DMMs and especially into the analysis-by-synthesis process. One of the reasons is that almost every model extension makes the model adaptation more complicated and a lot of methods rely on the rendering process to be differentiable. There is a dramatic gap between what current computer graphics or also deep learning-based image generation methods are capable of and what is state of the art for 3DMMs. Also generated instances usually lack facial details like wrinkles or moles which are challenging to render properly. Recent work aims at those challenges by using generative adversarial networks as texture models [Slossberg et al. 2018] but they are not modeled in the shape and not specially treated during rendering. A possible future direction is to either model or learn the gap between current 3DMM renderings and state of the art computer graphics or real-world 2D images.

An interesting open challenge is to better exploit the constraint of the 3DMM. Existing work uses generic pipelines for tasks such as rasterization or visibility calculation. However, the geometry is defined by a low dimensional parameter vector from which the per-vertex visibility could presumably be inferred more efficiently than treating the resulting mesh as a generic shape. The attempt of [Schneider et al. 2017] to learn the relationship between PRT coefficients and shape parameters is a first step in this direction.

5 ANALYSIS-BY-SYNTHESIS

3DMMs have been widely used for image-based reconstruction. Reconstructing a 3D face from an observed image(s) involves estimating the 3DMM coefficients which can best explain the observation. This is the inverse of the image synthesis process covered in the previous section.

Analysis-by-synthesis refers to a class of optimization problems which solves this by minimizing the difference between the observed image(s) and the synthesis of an estimated 3D face. Such an optimization problem can be ill-posed with several ambiguities and multiple minima. This is a widely researched problem, with a variety of solutions exploring different input modalities (Sec. 5.1), energy functions (Sec. 5.2) and optimization strategies (Sec. 5.3). We present publicly available approaches in Table 3.

Analysis-by-synthesis techniques have also recently been used in combination with deep learning architectures for learning-based reconstruction algorithms. We will discuss these methods in Sec. 6.

5.1 Input Modalities

Analysis-by-synthesis methods have been explored using multiple image modalities, from multi-view to monocular images and videos. While multi-view methods produce very detailed and high-quality results, capturing such data requires expensive setups. A lot of recent focus has been on obtaining similar quality reconstructions with much lower cost solutions, e.g., using a single RGB image. This has also led to an increase in commercial applications for the mass market. Fitting a 3DMM to 3D scans can also be considered as analysis-by-synthesis. This is related to registration techniques, covered in Sec. 3.

Multi-View Systems. We will start our discussion with multi-view solutions which minimize the photometric consistency between the multi-view images and the synthesis of the estimated reconstruction. Most multi-view methods, such as those covered in Sec. 2 do not require a strong prior in the form of 3DMMs. However, there are several methods which use 3DMMs to aid reconstruction in stereo camera systems. Model-based stereo reconstruction was explored in Wallraven et al. [1999]. The reconstruction quality was improved by eliminating the estimation of illumination and reflectance in Amberg et al. [2007]; Fransens et al. [2005]. 3DMMs also proved to be very valuable in low-resolution settings where high-quality image textures cannot be exploited, or under occlusions [Romeiro and Zickler 2007; Thies et al. 2018b]. Most of the methods discussed here solve very large optimization problems, and are not real-time. Thies et al. [2018a] is one real-time method which has a data-parallel implementation on a GPU.
<table>
<thead>
<tr>
<th>Publication</th>
<th>Input</th>
<th>Estimates</th>
<th>Approach</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Edge fitting</strong></td>
<td>2D image, landmarks</td>
<td>pose, shape</td>
<td>edge features, ICP</td>
<td></td>
</tr>
<tr>
<td>[Bas et al. 2017b]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Eos fitting library</strong></td>
<td>2D image, landmarks</td>
<td>pose, shape</td>
<td>landmark and contour fitting</td>
<td>Huber [2017] handles expressions</td>
</tr>
<tr>
<td>[Huber et al. 2016]</td>
<td></td>
<td></td>
<td></td>
<td>estimates posterior distribution, Egger et al. [2018] handles occlusion</td>
</tr>
<tr>
<td><strong>Basel Face Pipeline</strong></td>
<td>2D image, landmarks</td>
<td>pose, shape, expression, texture, illumination</td>
<td>MCMC Sampling</td>
<td></td>
</tr>
<tr>
<td>[Gerig et al. 2018]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Deep 3D Face Reconstruction</strong></td>
<td>2D image(s)</td>
<td>pose, shape, expression, texture, illumination</td>
<td>deep (ResNet)</td>
<td></td>
</tr>
<tr>
<td>[Deng et al. 2019]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>PRNet</strong></td>
<td>2D image</td>
<td>pose, shape</td>
<td>deep (convolutional)</td>
<td>outputs mesh in BFM topology</td>
</tr>
<tr>
<td>[Feng et al. 2018]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Expression-Net</strong></td>
<td>2D image</td>
<td>pose, shape, expression, texture</td>
<td>deep (ResNet)</td>
<td>bundles [Chang et al. 2017; Tran et al. 2017]</td>
</tr>
<tr>
<td>[Chang et al. 2018]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>RingNet</strong></td>
<td>2D image</td>
<td>pose, shape, expression</td>
<td>deep (ResNet)</td>
<td>handles occlusion</td>
</tr>
<tr>
<td>[Sanyal et al. 2019]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Pix2vertex</strong></td>
<td>2D image</td>
<td>pose, shape, expression</td>
<td>deep + shape from shading</td>
<td>shape beyond 3DMM</td>
</tr>
<tr>
<td>[Sela et al. 2017]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Facial Details Synthesis</strong></td>
<td>2D image</td>
<td>pose, shape, expression, appearance</td>
<td>UNet for details</td>
<td></td>
</tr>
<tr>
<td>[Chen et al. 2019]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>3DMMs as STNs</strong></td>
<td>2D image</td>
<td>pose, shape, expression</td>
<td>spatial transformer network</td>
<td></td>
</tr>
<tr>
<td>[Bas et al. 2017a]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>3D Face Reconstruction</strong></td>
<td>2D image, output of</td>
<td>shape details</td>
<td>estimate bump map using encoder-decoder architecture handles occlusions</td>
<td></td>
</tr>
<tr>
<td>[Tran et al. 2018]</td>
<td>[Tran et al. 2017]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>FLAME</strong></td>
<td>2D / 3D landmarks</td>
<td>pose, shape, expression</td>
<td>landmark fitting</td>
<td></td>
</tr>
<tr>
<td>[Li et al. 2017]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Basel Face Pipeline</strong></td>
<td>3D scan, landmarks</td>
<td>pose, shape, expression, texture</td>
<td>Gaussian process regression, nonrigid ICP</td>
<td></td>
</tr>
<tr>
<td>[Gerig et al. 2018]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>LSFM Pipeline</strong></td>
<td>3D scan</td>
<td>pose, shape, expression</td>
<td>nonrigid ICP</td>
<td>fully automatic</td>
</tr>
<tr>
<td>[Booth et al. 2016]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Model Fitting</strong></td>
<td>3D scan, landmarks</td>
<td>pose, shape</td>
<td>nonrigid ICP, template and model fitting</td>
<td>handles occlusions</td>
</tr>
<tr>
<td>[Brunton et al. 2014b]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Multilinear Model Fitting</strong></td>
<td>3D scan, landmarks</td>
<td>pose, shape, expression</td>
<td>nonrigid ICP, global model in Bolkart and Wuhrer [2015a], local model in Brunton et al. [2014a]</td>
<td>handles occlusions</td>
</tr>
<tr>
<td>[Bolkart and Wuhrer 2015a; Brunton et al. 2014a]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Monocular RGBD.** RGB-D sensors capture RGB as well as depth information of the scene. Consumer stereo cameras either use passive stereo, IR projection-mapping, or time-of-flight technology. The depth channel in the input helps in resolving depth ambiguities due to the lack of multiple views. Thus, in addition to photometric consistency, these methods also minimize depth consistencies using point-to-point and point-to-plane distances, see Sec. 3. Since monocular reconstruction methods solve a smaller optimization problem compared to multi-view methods, many real-time solutions exist [Bouaziz et al. 2013; Hsieh et al. 2015; Li et al. 2013; Thies et al. 2015; Weise et al. 2011a]. While most methods heavily rely on 3DMMs, some try to adapt them to capture user-specific details. [Weise et al. 2011a] build a user-specific expression model by adapting a general one. This is done in an offline stage before the online tracking. [Bouaziz et al. 2013; Li et al. 2013] adapt the 3DMM online, thus removing the need for an offline step. [Hsieh et al. 2015] introduced an occlusion robust tracking system using face segmentation masks. [Liang et al. 2014] reconstruct a single image by retrieving instances
of 3D shapes from a dataset and merging them, thus avoiding the need for 3DMMs.

Monocular RGB. Without the presence of the depth channel, the analysis-by-synthesis problem becomes even more ill-posed. These methods cannot easily resolve depth ambiguities. Thus, the prior knowledge of a 3DMM becomes important. Monocular RGB videos can provide more constraints. The identity component, in this case, can be estimated by fusing information from multiple frames in a preprocessing step. Many methods can track the face in real-time [Cao et al. 2015, 2014a, 2013, 2016a; Ichim et al. 2015; Thies et al. 2016]. As in the case of RGB-D based methods, there are methods which try to add details over the 3DMM reconstructions to make the results user-specific and detailed. [Garrido et al. 2016b] add medium-scale correctives based on spectral basis vectors. Cao et al. [2015]; Garrido et al. [2013, 2016b]; Shi et al. [2014]; Suwajanakorn et al. [2014] also add high-frequency wrinkle-level details. Wu et al. [2016b] use local blendshape models to capture more details compared to global blendshape based methods. [Cao et al. 2013, 2016a; Ichim et al. 2015] compute user-specific 3DMMs using images of a person performing specific known expressions.

Photo-collections, i.e., collections of images of a person can also be used to constrain the identity components of the reconstructions [Kemelmacher-Shlizerman and Seitz 2011; Liang et al. 2016; Piotrachke and Blanz 2016; Roth et al. 2015, 2016; Suwajanakorn et al. 2015]. This is a more unconstrained setting compared to multi-view images where all views are captured at the same time in the same environment. Approaches which use photo-collections and videos are more practical than multi-view images since such data is widely available for most people.

Reconstruction from a single image is the most challenging scenario. However, the original work of Blanz and Vetter [1999] already proposed an analysis-by-synthesis solution, see Fig. 5. While they required manual initialization for the optimization problem, several approaches made the approach more robust to enable automatic reconstruction [Adrian and Smith 2011a; Bas et al. 2017b; Egger et al. 2018; Fried et al. 2016; Hu et al. 2017b; Kortylewski et al. 2018c; Paysan et al. 2009b; Schneider et al. 2017; Schönborn et al. 2017; Tewari et al. 2018]. Most analysis-by-synthesis approaches evaluate the photometric consistency between the observations and the estimates. Some approaches have explored the use of other image features, such as edges, or SIFT [Booth et al. 2017; Romdhani and Vetter 2005], in order to obtain higher fidelity reconstructions. Occlusion robust reconstruction by jointly solving for segmentation has been explored in Egger et al. [2018]. Monocular reconstruction methods primarily differ in their formulated energy functions. We will look at these in detail in Sec. 5.2.

5.2 Energy Functions

The analysis-by-synthesis paradigm involves the solution of a nonlinear optimization problem made up of a number of energy functions. Methods differ in their combination and precise design of these energy functions, their relative weights and (dealt with in the following subsection) the optimization strategy used to minimize the energy. Here we describe the most commonly used energy functions for fitting to RGB images. For single image reconstruction, the energy functions are expressed in terms of a single set of unknown parameters $\Theta$. In the case of multi-view images of a static face, the camera parameters are indexed by viewpoint while all others are fixed across views. In the case of an image sequence of a dynamic face, camera, expression and lighting parameters are indexed by frame while neutral shape and texture parameters are fixed throughout the sequence.

Appearance error. The key ingredient of analysis-by-synthesis is to measure the difference between observed data and a synthesis using the model. Most directly, this is the appearance error between an input image and the rendered face. A number of variants of this term have been used. The pixel-wise formulation sums the appearance error over the pixels of the image, necessitating rasterization of the model:

$$E_{\text{appearance}}(\Theta, I_{\text{obs}}) = \sum_{(x,y)\in \text{foreground}} \| I_{\text{obs}}(x, y) - I_{\text{model}}^{\theta}(x, y) \|^2$$

(19)
where foreground = \{(x, y) \in I | \text{raster}_C \cap T \cdot \omega^s, \omega^r(x, y) \neq \text{NULL}\} is the set of pixels covered by the union of all triangles. This formulation naturally weights the contribution of model vertices in terms of their contribution the appearance of a pixel. An alternative is to compute the appearance error vertex-wise by rendering in model space and sampling the image intensities onto the vertices:

\[ E_{\text{vertex}}(\Theta, I_{\text{obs}}) = \sum_{j \in \text{visible}} \left\| \text{interp}[I_{\text{obs}}, \text{project}|C, v_j(w^s, w^e)] - I_\text{model}(\Theta) \right\|^2 \]

(20)

where interp[X, (x, y)] represents differentiable interpolation of 2D object X at location (x, y) and visible is the set of visible vertices. A common variant of this approach uses a random subset of vertices rather than all of them. This is more efficient, introduces stochasticity that may help avoid local minima and avoids overly conservative fits near the boundary where background may be sampled. Differentiable interpolation of the image can either be done with explicit differentiable sampling (e.g., bilinear sampling as in Jaderberg et al. [2015] or precomputing the image gradient and then interpolating this along with the image intensities (this was done in the original Blanz and Vetter [1999] paper). A drawback of the vertex-wise error is that regions of the image with dense coverage from projected vertices are weighted more heavily than more sparsely sampled regions. This can be overcome by using weights related to projected area. Blanz and Vetter [1999] accomplished the same effect by using the triangle area as the probability by which the triangle would be sampled at projected positions of vertices lying on model texture edges or the occluding boundary. Amberg et al. [2007] extended this to multiple views and improved robustness by averaging the cost surface over different parameters of the edge detector. Keller et al. [2007] showed that these cost functions are neither continuous nor differentiable. Bas et al. [2017b] transformed edge fitting into landmark fitting by alternating between computing an explicit correspondence between edge pixels and model edges and minimizing the resulting landmark energy. Sánchez-Escobedo et al. [2016] directly regress shape parameters from a set of multi-view occluding contours.

Finally, some other features have been considered. Romdhani and Vetter [2005] used the position of specularities in the image to constrain the surface normal direction at the corresponding location on the model via a specular reflection model. Booth et al. [2017] and Booth et al. [2018b] compute dense SIFT features from the input image and compare these to the SIFT features on which their statistical texture model is built in a similar fashion to the vertex-wise appearance error above.

Background Modeling. A common challenge when optimizing for pose and shape is the varying visibility of vertices for vertex-wise errors and the varying number of pixels covered by the face for pixel-wise errors. This leads commonly to the undesired effect of shrinking. Having the model covering fewer pixels or having fewer vertices visible leads to an undesired local optimum of most error terms. Common strategies to overcome this are fixed visibility, restrictive regularization, relying on landmarks, enforcing edge or contour terms or explicit image segmentation. Schönborn et al. [2015] demonstrated the problems with an implicit background model which is present in all error formulations and have shown that even simple background models \( b \) like a constant, a Gaussian or an image histogram-based model can solve this issue. The background model can easily be added to the existing formulations, e.g., for the pixel-wise formulation as:

\[ E_{\text{image}}(\Theta, I_{\text{obs}}) = E_{\text{appearance}}(\Theta, I_{\text{obs}}) + \sum_{(x,y) \in \text{background}} b(I_{\text{obs}}(x, y)). \]  

(22)

Occlusions and Segmentation. Occlusion of faces by other objects, that are not part of the generative model, are a common challenge for the so far presented error terms and for analysis-by-synthesis in general. There are various methods presented on how to identify occlusions. Those methods range from appearance-based methods [De Smet et al. 2006; Pierrard 2008] to detection [Morel-Forster 2016] and segmentation-based methods [Egger et al. 2018; Saito et al. 2016]. They share the basic idea, that occluded pixels are excluded from the model evaluation:

\[ E_{\text{semantic}}(\Theta, I_{\text{obs}}) = \sum_{l \in \text{label}} \sum_{(x,y) \in R(l)} E_{\text{pixel}}(\Theta, I_{\text{obs}}(x, y), l). \]

(23)
where each $E_{\text{pixel}}^{\text{label}}$ is a separate model per label, and $R(l)$ is the image region covered by label $l$. Those labels could e.g., be face, occlusion and background or also contain more detailed labels like beards. Whilst the segmentation is based on detection and fixed in Morel-\(\sigma\) where each $E_l$ vertex along with surface normals to satisfy a shape-from-shading constraint. This is regularized using the squared vertex distance between the updated shape and the closest shape in the 3DMM subspace enabling recovery of fine-scale detail not captured by the model. Allowing arbitrary shape or albedo changes transforms the problem into classical shape-from-shading and becomes highly ill-posed. For this reason, additional generic priors are used. Patel and Smith [2012] use a piecewise smoothness prior on per-vertex diffuse albedo which is allowed to vary per-vertex along with surface normals to satisfy a shape-from-shading constraint. This is regularized using the squared vertex distance between the updated shape and the closest shape in the 3DMM space. Richardson et al. [2017] use the same regularization, though, expressed in terms of per-pixel depth. To ensure smoothness, they also use the L1 norm of the discrete Laplacian of the depth map. The L2 norm of the mesh Laplacian has also been used as a smoothness prior [Garrido et al. 2016a; Tewari et al. 2018].

When reconstructing a dynamic face from video, parameters can either be assumed fixed (if identity dependent) or smoothly varying (pose, expression, lighting). These latter parameters can, therefore, be regularized with generic temporal smoothness priors. A common and simple way to express this prior is to initialize each frame with the estimate from the previous one. This encourages convergence to a local minimum close to the solution for the previous frame. More sophisticated priors have also been considered. For example, Cao et al. [2013]; Weise et al. [2011a] build a Gaussian mixture model over expression parameters from the previous $k$ frames. This model is then used to regularize the estimate for the current frame.

5.3 Optimization

From the perspective of optimizing the energy functions above, there are a number of significant challenges. First, most of the energy terms are nonconvex in theory and we observe in practice that there are many local minima. Second, the appearance error is not even continuous due to rasterization/vertex visibility and shadowing all being noncontinuous functions. Third, the appearance error has a small basin of convergence. When a model feature is completely misaligned to the image (or in the extreme case, the whole model aligned entirely to background), the gradient of the appearance error conveys no useful information. Fourth, all parameters have global influence. Fifth, computing the appearance error and its gradient is computationally expensive, amounting to the rendering of an image. For these reasons, a significant effort has gone into the selection of optimization algorithms and engineering of the optimization schedule to develop methods that are sufficiently fast and robust.

The majority of existing approaches optimize based on gradient information of the energy function. The original Blanz and Vetter [1999] approach used first-order gradient descent, as have other more recent methods [Bouaziz et al. 2013; Fried et al. 2016; Ichim et al. 2015]. Since they computed the appearance error over only a small subset of randomly selected triangles, this is strictly stochastic gradient descent (SGD). An interesting parallel here is that modern deep learning-based methods (see Section 6) are usually trained with SGD and use similar energy functions so they are learning from the same signal used in the original method.

Since the energy terms above can easily be formulated as nonlinear least-squares problems, specialized pseudo-second-order methods like Gauss-Newton or Levenberg-Marquardt have often been used [Garrido et al. 2013, 2016a,b; Romdhani and Vetter 2005; Thies et al. 2015, 2016]. Booth et al. [2017] use a “project-out” strategy in which appearance parameters are implicitly solved in a least squares sense and optimization takes place only over geometric parameters. General pseudo-second-order methods such as BFGS have been used [Cao et al. 2013; Weise et al. 2011a] as well as genuine second-order methods, specifically a stochastic variant of Newton’s method [Blanz and Vetter 2003]. As the problem size increases, as in the case of shape-from-shading, gradient descent becomes the most common optimization approach [Garrido et al. 2016a; Shi et al. 2014; Suwajanakorn et al. 2014; Tewari et al. 2018]. In all the above methods, the discontinuity of the appearance function is dealt with by fixing rasterization/visibility when computing gradients or even keeping them fixed for a certain number of iterations. Importantly, this means that the gradient cannot convey information about a change in visibility. Many other tricks have been considered, for example, hierarchical optimization (both in parameter space and spatially, i.e. multiresolution [Thies et al. 2016]) and using an optimization schedule in which different energy functions are switched on or weighted differently at different phases on the optimization [Blanz and Vetter 1999].

Several approaches have decomposed the energy terms into several smaller, often linear, problems (sometimes with closed-form solutions) that can be solved efficiently and in sequence [Aldrian and Smith 2010, 2011a, 2013, 2011b; Bas et al. 2017b; Cao et al. 2014a, 2013; Hu et al. 2017c; Romdhani et al. 2002; Saito et al. 2016; Zhu et al. 2015]. These alternating approaches are usually very efficient but not guaranteed to obtain the optimum solution that comes from optimizing all parameters simultaneously.

Gradient-based methods are typically initialized by fitting only to landmarks, i.e. to optimize the landmark energy in isolation. Originally, the landmark positions were provided manually but
while the albedo colors are similar for very different skin types with the 3DMM. Bas and Smith [2019] show how to express the [Tewari et al. 2018c] and the albedo illumination ambiguity [Egger 2018]. Ambiguities like the perspective face shape ambiguity to a manifold of possible solutions.

While gradient-based methods are widely used mainly due to computational efficiency and ease of implementation, these methods are sensitive to initialization and often end up in local minima. Probabilistic methods based on Bayesian inference were proposed to deal with these limitations [Egger et al. 2018; Kortylewski et al. 2018; Schneider et al. 2017; Schönborn et al. 2017]. These methods do not require any gradient computation of the energy terms to update the estimates. They are stochastic and thus, less susceptible to getting stuck in local minima. Different from optimization-based methods which only provide a single solution, these approaches approximate the full posterior distribution and thus provide access to a manifold of possible solutions.

5.4 Open challenges
Reconstructing 3D shape and albedo from a 2D image is an ill-posed problem. Ambiguities like the perspective face shape ambiguity [Smith 2016] and the albedo illumination ambiguity [Egger 2018] have been demonstrated (see Figure 6). These ambiguities can not be resolved completely and priors are our best approach to at least find a reasonable estimation. They are the major reason why there is a huge gap between the estimates we can get from multi-view and 3D data vs. from monocular images. Even in the state-of-the-art, it is often evident that overall skin color is explained using the lighting while the albedo colors are similar for very different skin types [Tewari et al. 2018]. This is somewhat improved by discriminative methods that do not need to synthesize the same appearance as a given image, only an image with the same identity [Genova et al. 2018], thereby sidestepping explicit estimation of illumination and camera parameters. Reporting the geometric errors obtained by the model mean is not common. Only three papers demonstrated their 3D reconstructions to be closer in mesh distance to the ground truth face compared to the model mean [Aldrian and Smith 2013; Sanyal et al. 2019; Schönborn et al. 2017].

Current state of the art techniques also lack dramatically in accuracy across pose and in terms of matching the contours and edges. It is very difficult to evaluate these beyond qualitative evaluation which makes it difficult to compare different approaches. Recently a first benchmark with natural images and ground truth shape was published and will help to better compare competing methods [Sanyal et al. 2019]. However, as methods get more accurate, the mesh distance errors get close to the range of error in computing “ground truth” using multi-view methods. This makes it difficult to quantitatively compare different approaches.

Another challenge which is usually neglected are occlusions. Faces are mostly occluded by objects which are frequently in front of faces like glasses, cigarettes, hands or microphones, but can also be occluded by virtually every other object. Analysis-by-synthesis methods fail when they do not explicitly model occlusions. Furthermore, reconstruction methods based on 3DMMs are limited to the space of faces covered by the models. A lot of residual error in the results stems from the fact that 3DMMs do not model detailed and high-frequency geometry and texture. Furthermore, most approaches use simple lighting models which cannot explain many in-the-wild images. These limitations are also shared with the learning-based methods which use analysis-by-synthesis in their pipeline, see Sec. 6.

Recent techniques have been focused on reconstruction from a single face individually. The aim of face image analysis would, however, go beyond interpreting a single face or each face separately. We would like to analyze and interpret interactions between people and perhaps also ease the analysis task by exploiting scene constraints, such as shared illumination parameters to deal with albedo-illumination ambiguity, or constraints on the perspective face shape ambiguity by analyzing multiple faces jointly.

6 DEEP LEARNING
So far, we have mainly discussed classical face modeling and parameter estimation techniques based on optimization-based inverse graphics. We now discuss how these processes can be replaced by or combined with deep learning, see Figure 7. There are a number of reasons for wanting to do this. On the modeling side, the use of nonlinear, deep representations offers the possibility to surpass classical linear or multilinear models in terms of generalization, compactness and specificity [Styner et al. 2003]. On the parameter estimation side, we can exploit the speed and robustness of deep networks to achieve reliable performance on uncontrolled images.

We begin by discussing deep modeling and deep model fitting before finally discussing methods that simultaneously learn both the model and how to fit it within a single deep network.
6.1 Deep Face Models

The traditional modeling techniques discussed in Section 3 aim to represent face shape, expression, and appearance as vector $\mathbf{w}$ in a low-dimensional latent space $\mathbb{R}^d$. The projection into (respectively reconstruction from) this latent space is defined by linear or multi-linear operations, and can be thought of as encoding (respectively decoding) the high-dimensional information in $\mathbb{R}^d$. Deep learning provides a new tool for building 3DMMs using nonlinearities both in the encoder and the decoder. This way of building morphable models is currently a very active area of research.

We can see the relationship between the encoder and decoder learned using deep learning and classical works using an example of linear models commonly used for shape and texture modeling. In the context of deep learning, such a linear model formalized in Equation (2), is exactly equivalent to a fully connected layer in a neural network. Concretely, the parameter vector $\mathbf{w}$ plays the role of the input features, the principal components $e_j$ are the weights and the mean $\bar{c}$ is the bias. This can be viewed as decoding from the latent parameter space to the data space $c$. Projection onto the model can similarly be viewed as encoding with a fully connected layer in which the input features are the data, the weights are the rows of the transposed principal component matrix and the biases are given by $-\bar{c} e_j^T$. Concluding the analogy, a PCA can be accomplished by combining the encoder and decoder as a linear autoencoder with a single hidden layer. Such an autoencoder with $d$ neurons in the hidden layer will learn a latent space with the same span as a $d$ dimensional PCA, though without the guarantee of orthogonality (though this could be ensured with appropriate loss functions).

Given this close relationship between classical methods and deep learning, it is natural to ask if there exist more powerful nonlinear models that can be trained based on current advances in deep neural networks. As in the classical work, this has been considered for the 2D case. Duong et al. [2019] propose a deep appearance model for 2D facial images that extends 2D AAMs to model nonlinearities. This is achieved using deep Boltzmann machines to model 2D shape and texture information. For modeling 3D faces, first successful models using autoencoders, GANs, and hybrid structures have been proposed, as detailed in the following.

Fernández Abrevaya et al. [2018] proposed the first encoder-decoder architecture to model the 3D geometry of faces. The encoder first projects the 3D face to a 2D image and uses a standard image-based encoder, while the decoder is fixed to a classical tensor-based face model. This allows decoupling shape variations caused by identity and expression. Bagautdinov et al. [2018] introduced a VAE that models different levels of detail of facial geometry by representing global and increasingly localized shape variations in different layers of the network. The 3D geometry is again represented using a two-dimensional mapping, and convolutions are performed in the image domain. This work allows representing highly detailed geometric information in latent space. Lombardi et al. [2018] extend this work to jointly encode variations in appearance and geometry, for the application of highly detailed facial rendering from novel viewpoints. Ranjan et al. [2018] proposed the first autoencoder architecture for the geometry of faces that performs convolutions in 3D mesh space directly instead of going through a 2D image representation. The model, named CoMA, allows for very compact representations of the facial geometry. This work was recently extended to encode both texture and shape information jointly [Zhou et al. 2019].

An alternative line of work considers learning GANs for 3D face modeling. Slossberg et al. [2018] proposed the first 3DMM using GANs. In this work, the facial texture is mapped to a coherent 2D image domain, and two-dimensional convolutions are employed to build a GAN of facial texture. This is combined with a standard PCA-based 3DMM for facial geometry, where for a generated face texture, a suitable PCA-based geometry is computed. Recently, multiple methods were proposed to generate 3D facial geometry, possibly with texture information. Fernández Abrevaya et al. [2019] proposed to train a GAN for the geometry of 3D faces that is able to decouple different factors of variation such as identity and expression. Shamai et al. [2020] proposed a GAN architecture to generate both facial geometry and texture, with a focus on highly detailed texture information by mapping the face to a unit rectangle. Cheng et al. [2019] proposed the first intrinsic GAN architecture that operates directly on 3D meshes. As in the case of 2D images, GANs are generally able to generate more detailed and realistic 3D faces than autoencoders at the cost of being more difficult to train.
Finally, hybrid structures can be effective to learn nonlinear 3DMMs. Tran and Liu [2018a] jointly learn a 3DMM and 3D reconstruction from a 2D image using a differentiable renderer in the training loss, see also Section 6.3. The network takes as input a 2D image and encodes it into projection, shape and texture parameters. Two decoders are then used to infer 3D shape and texture, respectively. Wang et al. [2019b] proposed an adversarial auto-encoder structure that allows disentangling factors of variation such as identity, expression, or pose of 2D facial images, and that is trained in an unsupervised way. While the method’s input and output are 2D images, the 3D geometry of the face can be reconstructed.

Recently, appearance modeling approaches based on deep learning have also been proposed. The rise of deep learning methods facilitated to learn per-vertex appearance models directly from images, such as done by Tewari et al. [2018], who learn per-vertex albedo model offsets in order to improve the generalization ability of an existing PCA-based model. Similarly, Tewari et al. [2019], learn a per-vertex albedo model from scratch based on video data. Zhou et al. [2019] train a mesh decoder that jointly models the texture and shape on a per-vertex basis, which, however, relies on the availability of 3D shape and appearance data. There are also several deep learning approaches that consider a texture-based appearance modelling. Without the need of 3D data, Tran and Liu [2018a] learn a nonlinear facial appearance model represented in uv-space based on CNNs, which, however, does not explicitly consider lighting. In follow-up work, the authors considered a more elaborate model where the albedo and the lighting is separately modeled [Tran et al. 2019; Tran and Liu 2018b]. Moreover, a range of generative methods that synthesize facial textures have been proposed, e.g., by Saito et al. [2017], Slossberg et al. [2018], Deng et al. [2018], Lombardi et al. [2018], Nagano et al. [2018] and Yamaguchi et al. [2018]. Gecer et al. [2019b] use GAN-based texture model for the task of 3D face reconstruction, and Nagano et al. [2019] use GAN-based texture models for the task of face normalization.

6.2 Deep Face Reconstruction

In the following, we discuss dense monocular face reconstruction approaches that are based on deep neural networks. We discuss requirements on the used training data, as well as different training strategies. Let us first have a closer look at the reconstruction problem. Blanz and Vetter [1999] tackle monocular face reconstruction by fitting a parametric model based on an optimization approach, i.e., gradient descent. Deep learning approaches follow a similar optimization strategy, but instead of solving the optimization problem at ’test’ time, they for example train a parameter regressor based on a large dataset of training images, see Figure 7. The regressor can be interpreted as an encoder network that takes a 2D image as input and outputs the low-dimensional face representation. Learned encoders can be combined with decoders based on classical face models to give rise to end-to-end encoder-decoder architectures. This methodology is widely-used and enables the fusion of classical model-based and deep learning approaches.

6.2.1 Supervised Reconstruction. Supervised regression approaches are trained based on paired training data, i.e., a set of monocular images and the corresponding ground truth 3DMM parameters. One of the essential questions here is how to efficiently obtain the ground truth for such a supervised learning task. In the following, we will categorize the approaches based on the type of employed ground truth training data.

One option would be to let users annotate the ground truth. While this is a popular strategy, which is often employed for sparse reconstruction problems [Saragih et al. 2011], the accurate annotation of dense geometry, appearance, and scene illumination is almost intractable. A related approach is for example employed in the work of Olszewski et al. [2016], where three professional animators manually created the blendshape animation to match a video clip.

For dense reconstruction tasks, some approaches [Laine et al. 2017] are trained based on images captured in a controlled multi-view capture setup. Thus, ground truth can be obtained by a multi-view reconstruction approach followed by fitting a 3DMM to the resulting 3D data. Normally, the ground truth is of very high quality, but the distribution of the captured monocular images does not match in-the-wild data, which can lead to generalization problems at test time.

The approach of Tran et al. [2017] performs monocular reconstruction for multiple images of the same person and computes a consolidated face identity based on simple averaging of the 3DMM parameters.

Currently, many approaches [Feng et al. 2018; Kim et al. 2018b; Klaudiny et al. 2017; McDonagh et al. 2016; Richardson et al. 2016; Sela et al. 2017; Yu et al. 2017] in the research community are trained on synthetic training data, since it is easy to acquire and comes by design with perfect annotations. Given a face 3DMM, random identities and expressions can be sampled in parameter space. Afterward, the models can be rendered under randomized illumination conditions and from different viewpoints to create the monocular images. Often, background augmentation is employed by rendering the generated faces on top of a large variety of real-world background images. Since all the parameters are controlled, they are explicitly known and can be used as ground truth. While it is easy to get access to synthetic training data, there is often a large domain gap between synthetic and real-world images, which severely impacts generalization to real images. For example, hair, facial hair, torsos, or mouth interiors are often not modeled at all. One possibility to counteract this problem in the future would be better models that include all these components.

To leverage the advantages of both real as well as synthetic training data, many current approaches [Kim et al. 2018b; Richardson et al. 2017] are trained on a mixture of data from these two domains. The hope here is that the approach learns to deal with real-world images, while the perfect ground truth of the synthetic training data can be used to stabilize training. One interesting variant of this is self-supervised bootstrapping [Kim et al. 2018b] of the training corpus. Other approaches that can be trained without requiring ground truth data are presented in the next sections.

6.2.2 Self-Supervised Reconstruction. Supervised training of a convolutional neural network requires an annotated dataset. Most of the methods we have discussed so far use such datasets, either synthetic or real. Recently, some approaches explored self-supervised learning i.e., training on real image datasets without any 3D labels.
Applying deep learning to the analysis of 3D face data is an active research topic. Tewari et al. [2017] introduced a model-based encoder-decoder architecture, which replaces the trainable decoder with an expert-designed fixed decoder. This expert-designed decoder takes the 3DMM parameters (latent code) predicted by an encoder as input and transforms them into a 3D reconstruction using the 3DMM. It further renders a synthetic image of the reconstruction using a differentiable renderer. Extrinsic parameters required for rendering are also predicted by the encoder. The loss function used is very similar to those used in analysis-by-synthesis (Sec. 4), consisting of photometric alignment and statistical regularization. We can think of such a technique as a joint analysis-by-synthesis optimization problem over a large training dataset, instead of a single image, see Figure 7. This allows for training a parameter regressor without any 3D supervision. This concept, usually in combination with supervised synthetic data has also been explored using higher-level loss functions like identity preservation [Genova et al. 2018; Sanyal et al. 2019], or perceptual and adversarial losses [Tran et al. 2017]. Gecer et al. [2019b] employ GANs in combination with differentiable rendering to learn a powerful generator of facial texture. [Richardson et al. 2017; Sengupta et al. 2018] refine 3DMM predictions for higher quality or more detailed results. [Deng et al. 2019; Sanyal et al. 2019] extend the network architecture to allow for training using multiple images of a person as constraint. Bas et al. [2017a] use a 3DMM as a spatial transformer network such that model fitting is learned as a by-product of solving a downstream task.

6.3 Joint Learning of Model and Reconstruction
Model-based encoder-decoder networks consist of a trainable encoder and a fixed decoder, where the decoder implements a 3DMM. However, the 3DMM itself could be trainable. We could simply update its values using the gradients from the loss function. This would allow face model learning using only 2D supervision. Learning 3D models entirely from 2D data was first shown in [Cashman and Fitzgibbon 2012] without the use of deep learning. Several deep learning approaches have explored refining an existing 3DMM using large image datasets [Lin et al. 2020; Tewari et al. 2018; Tran et al. 2019; Tran and Liu 2018b]. Nonlinear convolutional decoders have also been used to build nonlinear face models [Tran et al. 2019; Tran and Liu 2018b]. Models learned from 2D data are more generalizable to different identities, as the image datasets contain significantly more identities compared to the 3D datasets used to compute 3DMMs. Recently, an extension of the model-based encoder-decoder architecture was used to learn the identity component of a face model from videos [Tewari et al. 2019].

6.4 Open Challenges
Applying deep learning to the analysis of 3D face data is an active research topic that the community has only started to explore during the past few years, with many ongoing advances. Hence, many challenges currently remain to be solved. The most pressing ones include analyzing the limitations of current methods and providing comprehensive comparisons. This includes a clear analysis of the methods’ tendency to overfit, especially when mostly synthetic data is used for training and the interpretability of the learned representations. It also includes a clear analysis of whether training in the 2D or 3D domain offers clear benefits for different applications.

It is interesting that deep learning methods are learning from essentially the same energy functions as classical methods using similar optimization approaches (e.g., stochastic gradient descent). The difference is that backpropagation updates are averaged over batches and whole datasets, seemingly alleviating problems of local minima or overfitting to a single sample. The problem then becomes overfitting to the distribution of faces in the training set. The training data used in these learning-based methods are often biased (e.g., Liu et al. [2015] includes mostly smiling faces). This leads to biases in the reconstruction methods. A practical question that requires to be solved is to determine the minimum amount of data required to apply deep learning methods. This is important when high-quality data is used for supervised training.

As learning-based and analysis-by-synthesis methods come together through self-supervised reconstruction methods, there are many shared challenges such as perspective face shape ambiguities and dealing with occlusions (e.g., Tran et al. [2018] already did a first step in this direction). Learning-based methods typically are very fast and robust to initialization but achieve lower quality results compared to analysis-by-synthesis methods. One way to combine the desirable properties of these different paradigms is to use the learning-based solution as initialization for analysis-by-synthesis optimization [Tewari et al. 2018].

While some recent methods have tried to build 3DMMs just from 2D data for better generalization, the resulting models are not as high-quality and lack details due to the low resolution of faces in currently available in-the-wild images. Bridging the gap in terms of details between models trained using high-quality data, and those built using only 2D data is an important open challenge.

Other challenges include extending recently developed methods to new applications. For instance, while monocular face reconstruction has started being explored, there is not yet much work on reconstructing a coherently deforming facial geometry from 2D video data.

7 APPLICATIONS
Parametric face models enable many compelling applications. In the following, we will discuss applications in the domains of face recognition, entertainment, medical applications, forensics, cognitive science, neuroscience, and psychology. All these applications have been pushed by the availability of publicly shared models and code (see Tab. 2), as well as other resources [Community 2019].

7.1 Face Recognition
In the context of face recognition, 3DMMs have a manifold of potential applications. Blanz et al. [2002] proposed to perform face recognition using the cosine angle on the shape and color coefficients estimated from a pair of 2D images as a distance metric for identification and recognition. This distance metric exploits the natural disentanglement of 3DMMs separating identity (shape and color) from camera and illumination variation. It was shown that this 3DMM-based distance metric enables to recognize faces across
large pose and illumination variations [Blanz et al. 2002; Blanz and Vetter 2003; Paysan et al. 2009a], while being robust to facial expressions [Gerig et al. 2018], as well as being able to do recognition from features in the facial texture [Piepraz and Vetter 2007]. Recently, Tran et al. [2017] have shown that the performance of face recognition with 3DMM parameters can be enhanced by specifically taking the face recognition task into account when regressing the 3DMM parameters. Whilst most work focuses on face recognition from 2D images, the 3DMM was also applied to the 3D face recognition task focusing on the shape coefficients and robust recognition with respect to facial expressions [Amberg et al. 2008; Paysan et al. 2009a; ter Haar and Veltkamp 2008].

Although 3DMMs have shown promising results at face recognition in controlled settings, they did not achieve a convincing performance on in the wild data. This arises from the ill-posed problem of estimating shape and color parameters from a 2D image, at the same time a high precision of this estimation is needed for face recognition. Therefore, purely data-driven approaches have remained the dominant approach to face recognition, in particular since the advancement of deep learning technology [Parkhi et al. 2015; Schroff et al. 2015; Taigman et al. 2014]. However, data-driven approaches have fundamental problems such as their dependence on large-scale training data and their lack of generalization to out-of-distribution samples [Klare et al. 2012]. One of the main issue for face recognition is the alignment of images. Careful alignment of face images has a big impact on face recognition accuracy and even for state of the art deep learning systems. 3DMMs are particularly useful for tackling these limitations, e.g., by using 3DMMs as a tool for face frontalization [Blanz et al. 2005; Hassner et al. 2015; Tena et al. 2007]. In this context, it was shown that 9 out of 10 2D algorithms in the Face Recognition Vendor Test 2002 [Phillips et al. 2003] improved considerably when combined with a 3DMM for face frontalization [Blanz et al. 2005]. Other applications of 3DMMs include augmenting real-world data in 3D [Masi et al. 2016] and the generation of synthetic data for training [Kortylewski et al. 2018b; Sela et al. 2017] and for analyzing the effects of dataset bias on face recognition systems [Kortylewski et al. 2018a, 2019].

Almost all applications of 3DMMs in the context of face recognition would benefit from improvements of the parametric model as well as the fitting process. A more realistic texture model including textural details and modeling hair would enhance the quality of synthetic data, possibly further reducing the amount of real-world data needed to train data-driven models. A more accurate fitting process would enhance the model’s performance on face frontalization and face recognition from 3DMM parameters.

7.2 Entertainment

3DMMs are an integral building block for many compelling applications in the entertainment sector. Such applications normally have to work in the wild and based on a low number of sensors, e.g., only the images captured by a single color camera are accessible. In such underconstrained scenarios, the statistical prior that is encapsulated in the face model is a powerful tool to better constrain the underlying reconstruction problems. In the following, we discuss several entertainment applications in detail. These applications are also covered in more depth in the state of the art report of Zollhoefer et al. [2018].

7.2.1 Controlling 3D Avatars for Games and VR. Realistic 3D face avatars can be reconstructed based on multi-view video [Lombardi et al. 2018], a few images [Cao et al. 2016b; Ichim et al. 2015] or given only a single image [Hu et al. 2017b; Wang et al. 2019a]. Such avatars or even artist-designed characters can be controlled in gaming scenarios based on dense trackers that employ a parametric face model. Such vision-based control was first demonstrated in an off-line setting [Chai et al. 2003; Chuang and Bregler 2002; Pighin and Lewis 2006; Wang et al. 2004; Weise et al. 2009]. Nowadays, dense facial performance capture is feasible at real-time rates based on RGB-D [Li et al. 2013; Thies et al. 2015; Weise et al. 2011b] and color [Bouaziz et al. 2013; Thies et al. 2016] cameras. Besides vision-based animation, there is extended work on audio-based control [Cudeiro et al. 2019; Karras et al. 2017; Kshirsagar and Magnenat-Thalmann 2003; Taylor et al. 2017]. Face tracking can also be used to enable face-to-face communication [Li et al. 2015a; Lombardi et al. 2018; Olszewski et al. 2016; Thies et al. 2018c] in virtual reality.

7.2.2 Virtual Try-On and Make-Up. Face reconstruction and tracking based on a parametric face model can also be employed to build virtual mirrors that enable the try-on of accessories or make-up.
To this end, first, a personalized model of the face is recovered and tracked across the video resulting in a dense set of correspondences. These enable spatio-temporal re-texturing, e.g., to virtually place tattoos [Garrido et al. 2017] and can be used to add facial make-up [Bronstein et al. 2007] and try out different suggestions [Scherbaum et al. 2011]. Virtual make-up can be applied based on a reflectance/shading decomposition [Li et al. 2014b, 2015b]. Similar techniques enable the try-on of accessories, e.g., eyeglasses [Azevedo et al. 2016; Niswar et al. 2011].

7.3 Medical Applications

The clinical applications of the 3DMM cover both, analysis as well as synthesis. The dominant applications lie in analysis, where diseases can be recognized by facial shape. One example of such an effect is the classification and early diagnosis of fetal alcohol spectrum disorder [Suttie et al. 2013] or epilepsy [Ahmedt Aristizabal 2019]. Similarly, Hammond et al. [2004] demonstrated both visualization and recognition of congenital craniofacial growth disorders. Both these works used 3D data. However, the capability of 3D reconstruction from 2D images was explored for the screening of acromegaly [Learned-Miller et al. 2006] and genetic disorders [Tu et al. 2018].

In the direction of synthesis, the 3D shape model was explored to perform reconstruction of missing face parts based on the model statistics [Basso and Vetter 2005; Mueller et al. 2011]. Such a reconstruction can be applied for personalized implant design. Another work explored the synthesis capabilities for analysis and generated controlled stimuli to study responses in the fusiform face area and correlates them with autism spectrum disorder [Jiang et al. 2013]. 3DMM and statistical shape models, in general, are a popular standard framework in the field of medical imaging for segmentation and as models of variations in anatomical structures [Zheng et al. 2017]. A lot of those applications deal with pathologies in young or elderly people which are underrepresented even in the biggest face models [Ploumpis et al. 2019]. Those applications would profit from models built from a wider population or models than can better generalize beyond the data they are trained on.

7.4 Forensics

Applications in forensics range from identikit pictures over virtual aging to face reconstruction from dry skulls and recently also the detection of manipulated videos.

Describing faces from vague mental images is a challenging task. A tool based on a 3DMM [Blanz et al. 2006] allows exploring correlations within the face to generate indentikit pictures when providing descriptions based on vague features.

Virtual aging is a challenging task and can be helpful to later find missing children or victims of sexual abuse. The 3DMM helps to reduce the subjectivity of age progression methods. Several works in this direction are modeling age trajectories on 3DMM shape [Hutton et al. 2003; Koudelová et al. 2015; Shen et al. 2014] and at least two attempts have been made to do so for both 3DMM shape and texture [Hunter and Tiddeman 2009; Scherbaum et al. 2007]. Most methods focus on children and neglect textural details or wrinkles which are modeled in Pascal [2010]; Schneider et al. [2019].

Face reconstruction from dry skulls is an ill-posed problem. The mapping from the skull to face is not a one-to-one, but a one-to-many mapping. Models allow to control attributes for this reconstruction [Paysan et al. 2009b], explicitly estimate the posterior solution of possible faces per skull [Madsen et al. 2018] or model soft tissue thickness directly grounded by a 3DMM [Gietzen et al. 2019].

Recently 3DMMs were used successfully to generate or manipulate images and videos as discussed in Chapter 7.2.4. At the same time 3DMMs are also helpful to detect those manipulations from state of the art methods with high accuracy [Rossler et al. 2019].

7.5 Cognitive Science, Neuroscience, and Psychology

The ability to generate faces that can be controlled via parameters is very popular when studying how the human and non-human primate brain process faces. Studies with generated stimuli from a 3DMM can be found in Cognitive Science, Neuroscience, Psychology, and Social Science.

One of the earliest works using 3DMMs presented high-level aftereffects that indicate a model related to a statistical face model.
in the human brain. Those aftereffects were demonstrated using caricaturized faces and antifaces [Leopold et al. 2001]. Later it was shown that those results can not only be observed as aftereffects but also as responses of single neurons across caricaturization in macaque monkey to principal axes of a 3DMM [Leopold et al. 2006]. Later those aftereffects were shown to incorporate 3D information [Jiang et al. 2009a]. The effects based on caricatures for recognition were recently also investigated with 3DMMs in artificial neural networks trained on face recognition [Hill et al. 2018].

A topic that was heavily researched over the past decades and is still under investigation is how much the 3D shape contributes to face perception and if the face representation in our brain is built as a 3D model. Early studies based on functional MRI and behavioral techniques evaluated a shape-based model of human face discrimination [Jiang et al. 2006]. Later studies investigated the importance of 3D shape and surface reflectance [Jiang et al. 2009b] and event-related potentials to 3D shape are faster than to surface reflection [Caharel et al. 2009]. Other work explored how well humans can estimate a profile picture from a frontal view [Schumacher and Blanz 2012].

Recently it was shown that a face-processing system based on stepwise inverse rendering correlates better to neural measurements in macaque monkey than state of the art artificial neural networks [Yildirim et al. 2020].

Face image manipulation is another key application of the 3DMM to generate stimuli [Walker and Vetter 2009] to e.g., investigate social judgments based on facial appearance. Again the ability to control exactly what is manipulated is key for those research results sometimes measuring subtle effects [Walker et al. 2011]. Recently a dataset of controlled manipulated images was released to perform such experiments [Walker et al. 2018].

One of the major limitations compared to 2D based methods is that 3DMMs do not include hair. In a lot of studies faces and hair are not separated since faces without hair appear less face-like. For those models, it plays a substantial role to have controls over the parameters and that parameters can be interpreted which secures the future of 3DMMs in those fields.

## 8 PERSPECTIVE

In this last section, we want to look beyond the state of the art. We explicitly highlight the unsolved challenges in the field. In addition to focusing on face models, we look further and share our thoughts about the scalability of 3DMMs beyond faces. We also share our thoughts of the applicability of models including data, model and algorithm sharing also with its potential of misuse. We close with an outlook on how a 3DMM could look like in 10 or 20 years.

### 8.1 Global Challenges

In this section, we summarize the major open challenges that are shared across the different parts of 3DMMs. Local challenges that are specific to capturing, modeling, image formation or analysis-by-synthesis are mentioned in the respective sections.

One of the leading challenges is the balance between a low-dimensional parametric model and the degree of detail we are capable of modeling. Parametric models for eyes, teeth, hairs, skin details, soft tissue or even anatomical grounded muscles are not available. Additional complexity also renders analysis-by-synthesis even more challenging. Building faces with all those details is currently possible for a single face with a lot of manual labor, but automatic methods to extract those details or build models on top of them are in their beginnings. Current state of the art methods from capture, to modeling over image formation to analysis-by-synthesis use a lot of oversimplifying assumptions. Besides including more facial details there are also models that exploit the knowledge that a face is part of the body. Whilst faces and bodies are mostly analyzed separately, there exist first models that include faces and bodies jointly [Joo et al. 2018, Pavlakos et al. 2019]. Pavlakos et al. [2019] presented first results indicating that fitting the whole body is also beneficial for the quality measured in the face region only. Another major challenge is the comparability of all the components of a 3DMM. Already the modeling itself can only be evaluated on specific tasks and different models have a different focus and might perform better on a specific task. For analysis-by-synthesis, comparing the performance of a model and also of the model adaptation algorithm is an unsolved problem. Current state of the art research frequently focuses on task-specific qualitative results and those results can barely be compared across models and algorithms. The current trend in the community to share source-code and models helps to compare and reproduce results, however, there is a lack of useful benchmarks. A first step in this direction is a new dataset providing natural images in combination with a 3D scan of the same individual [Sanyal et al. 2019]. However this is focused on face reconstruction only, there is no single benchmark for 3D reconstruction from 2D images including illumination and albedo estimation.

The last challenges are of an ethical nature. Concerns around image analysis and synthesis, especially for faces is currently discussed within the scientific community as well as in the media and the broad public. The current algorithmic development in computer vision and graphics allows to recognize faces and to generate or manipulate images and video. In addition most methods around 3DMMs elicit some dataset bias. Saito et al. [2017] approached this using the Chicago Face Database [Ma et al. 2015] to build a face model with balanced ethnicities. Those challenges are not a purely scientific one, but also a political one. We start to see regulations of those technologies and there will be likely more regulations across the world in the near future. As a community, we can choose on what projects we focus to work on and there are plenty of meaningful and valuable applications of 3DMMs, face analysis, and face synthesis as we presented in the Section 7 which could be explored less with restrictive regulations.

### 8.2 Scalability

Research on parametric models of human faces has seen a lot of progress in recent years. This raises the question of how scalable the found solutions are to other types of real-world entities beyond humans. On the one hand, human faces are highly challenging as we are attuned to noticing even slightest inaccuracies in their modeling. At the same time, they are also more amenable to statistical modeling as their structure is relatively regular and correspondence
across faces is quite well-defined. Other types of real-world entities, or even humans in clothing or the human head with full hair, are exhibiting much stronger appearance, structure, and shape variation that may require additional methodical innovations to empower proper modeling. The vision and graphics communities have begun to build and learn statistical models of other types of shape categories. Researchers also increasingly attempt to learn such models in an unsupervised or weakly supervised way for better real-world scalability. These approaches partially build on many concepts learned from the models described in this article but introduce additional representation innovations, like learned implicit representations [Cole et al. 2017; Eslami et al. 2018; Sitzmann et al. 2019], to handle their specific structural properties. Future research will certainly see more work in this direction that answers the question of what is the right shape, appearance and deformation representations for a wider range of real-world object classes.

8.3 Application

An additional challenge for our research community will be to agree on efficient ways to share and combine research efforts performed by different research groups. We should agree on common data formats and dissemination channels for available scan databases, which would simplify building integrated models, and enable us to better test and compare them. In that context, ever more pressing questions of privacy and security will also need to be addressed. On the one hand, it is needless to say that we have to adhere to highest standards of privacy protection in data sets we share, so not to reveal personal data or identities beyond what is needed and permitted by law or by the captured individuals. For handling this, community-wide procedures for providing consent on the use of data that are compatible with legal regulations could be agreed on and shared.

However, beyond this, increasingly powerful methods to build and reconstruct such face models from image and video will in the future enable us to build highly believable 3D human avatars from casually captured imagery. These avatars will enable us to create virtual renditions of real people at unseen accuracy to populate computer graphically generated virtual spaces at high visual fidelity. However, algorithmic tools should be investigated as well that prevent the reconstruction or use of such avatars in undesired or questionable applications that a reconstructed person did not provide consent on. Advanced reconstruction algorithms on the basis of parametric models may also make it possible to extract semantic information of people from imagery that they may not want to reveal (e.g., about emotional state, health, and physical condition, etc.). Therefore, algorithmic strategies to balance personal privacy and reconstruction ability shall be investigated and provided by our research community.

Also, the continuously improving performance of algorithms to reconstruct detailed human models from single images or videos enables advanced new ways to synthesize new face imagery or even modify existing face images and videos at very high visual fidelity. As an example, some recent combinations of model-based reconstruction algorithms and adversarially trained neural networks have shown impressive results in that respect. Such advanced synthesis algorithms will simplify many applications and open up entirely new applications, for instance in content creation for animation and visual effects, in content creation for virtual and augmented reality, in telepresence, visual dubbing or advanced video editing. However, they might also be used to create or modify media content with malicious intent. Therefore, as a community focusing on basic research, we will continue our efforts to objectively inform the general public about the great possibilities opened up by advanced parametric models of face, body and other real-world entities to build the next generation of intelligent, interactive and creative computing systems. At the same time, we will use our essential basic expertise about the underlying algorithmic principles to develop new ways to detect unwanted media synthesis and modification and to prevent such unwanted modifications algorithmically.

8.4 Outlook

The big question we ask is how will a generative face model look like in 10 or 20 years? What will be the representation and will it be a complete model of the human face with all its variation and details? Currently, we experience a divergence of 3DMMs. Different research teams put a different focus and model some parts in more detail but lack other details or statistical variation. Recent modeling advances are focusing on building task-specific representations rather than a more general face model to be applicable for multiple tasks. For some applications, the model itself is the limiting factor, whilst other applications profit from a simple model based on PCA. The requirements in terms of quality, realism, generalization, and performance are very different e.g., content creation vs. computer vision. The gap between state of the art computer-generated renderings for a single face including expressions versus generative and parametric face models based on statistics is dramatic.

Current advances in the field of machine learning will contribute to build more general and at the same time more realistic models. The core of the face model was always interpreted as a learning problem, recent advances lifted the analysis-by-synthesis task from a per image optimization task to a learning challenge. However, this loop is not yet closed – why not learn or improve the model itself? There are already first works in the direction of model learning (compare Section 6.3), but they are limited by very similar modeling assumptions as traditional 3DMMs. First steps to overcome those were recently performed in the direction of neural rendering [Eslami et al. 2018; Thies et al. 2019], 3D representation learning [Sitzmann et al. 2019] and unsupervised shape model learning [Szabó et al. 2019]. Other modeling approaches like generative adversarial networks [Goodfellow et al. 2014; Karras et al. 2019b,a] are currently operating in 2D image space. Such parametric models can be used to embed faces of real people in a latent space Abdal et al. [2019a,b], but the resulting embedding is hard for humans to interpret.

20 years ago 3DMMs were part of a revolution in computer graphics and computer vision to go away from 2D image processing to 3D modeling. The computer vision community is currently focusing again on mainly 2D based approaches and we have to propose the missing key to again move the community to 3D. Additionally one of the leading benefits of 3DMM is the natural disentanglement of
shape, color, illumination and camera parameters. Such a disentanglement is very hard to be derived purely from data [Locatello et al. 2019] and for faces, 3DMMs build it manually based on the image formation process. According to “Pattern Theory” [Grenander 1996; Mumford and Desolneux 2010] it is a prerequisite for any high-performance image analysis system to find and separate conditional independent parameters that describe the image to analyze. The discovery and separation of such parameters purely from 2D data is still an unsolved challenge. 3DMMs directly implement models using the parameter also used by physics and geometry to model light and three-dimensional objects.

One direction which might be particularly interesting is to break out of the common modeling assumptions and oversimplification but at the same time automate the tedious manual work behind the photo-realistic generation of faces. We expect some kind of living 3DMM to evolve from the community. Automation will be the leading modeling idea. A living 3DMM should be able to learn from 3D data as well as 2D data, both still and in motion. We imagine the model to be learned from a minimal seed like a mean face, a sphere or just a rough prototype based on the first few data points. The optimal living model would not be task-specific but should be able to generalize to various tasks. The face model must, therefore, be hierarchical in some form to represent multiple degrees of detail but share statistics across those levels. Such an optimal face model would be general enough to be applicable for real-time computer vision tasks, analysis-by-synthesis from currently challenging images as well as photorealistic rendering with a high level of facial details. Last but not least some tasks rely on an interpretable parametrization and not just a black box learning machine. Basic knowledge of geometry and physics would not only ease the learning but also at least disentangle pose and illumination variation from the facial shape and appearance. Building such a general face model might remain a challenge for the next 10 or 20 years but would align with the original idea behind 3DMMs.

ACKNOWLEDGMENTS
This survey paper was initiated at the Dagstuhl Seminar 19102 on 3D Morphable Models [Egger et al. 2019] and contains ideas resulting from discussions at this seminar. This survey paper was partially funded by Early PostDoc Mobility Grant, Swiss National Science Foundation P2BSP2_178643, ERC Consolidator Grant 4DRePLy and the Max Planck Center for Visual Computing and Communications (MPC-VCC). We thank Barış Geçer for his help on the teaser figure, and Haiwen Feng for providing the FLAME texture space. We thank the anonymous reviewers whose comments have greatly improved this manuscript.

REFERENCES


3D Morphable Face Models - Past, Present and Future