Semantic Deep Face Models

Prashanth Chandran[1,2], Derek Bradley[2], Markus Gross[1,2], and Thabo Beeler[2]

[1]Department of Computer Science, ETH Zürich
[2]DisneyResearch|Studios, Zürich
chandrap@inf.ethz.ch, derek.bradley@disneyresearch.com, grossm@inf.ethz.ch,
thabo.beeler@gmail.com

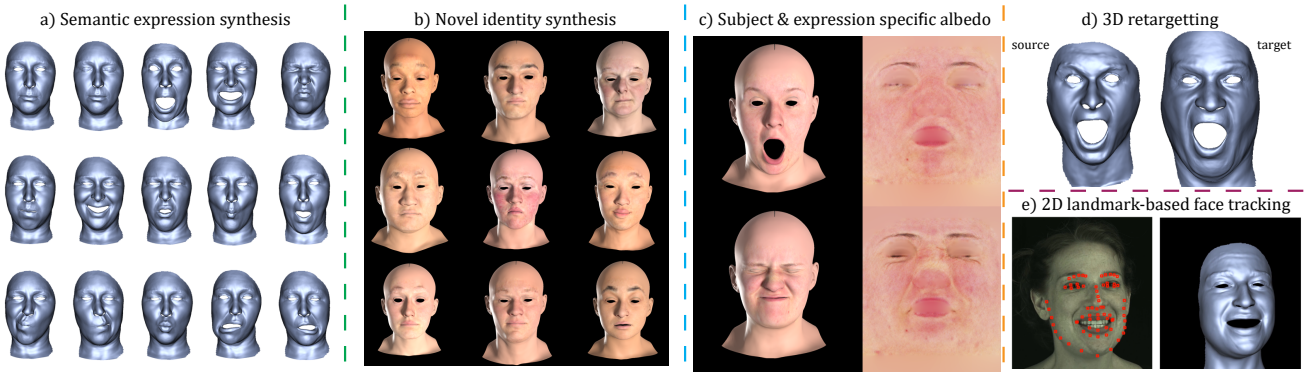


Figure 1: We propose semantic deep face models—novel neural architectures for modelling and synthesising 3D human faces with the ability to disentangle identity and expression akin to traditional multi-linear models. We demonstrate several applications of our method including (a) semantic expression synthesis, (b) novel identity synthesis (c) generation of expression specific high resolution albedo maps, (d) 3D facial performance retargeting, and (e) 2D landmark based face tracking.

Abstract

Face models built from 3D face databases are often used in computer vision and graphics tasks such as face reconstruction, replacement, tracking and manipulation. For such tasks, commonly used multi-linear morphable models, which provide semantic control over facial identity and expression, often lack quality and expressivity due to their linear nature. Deep neural networks offer the possibility of non-linear face modeling, where so far most research has focused on generating realistic facial images with less focus on 3D geometry, and methods that do produce geometry have little or no notion of semantic control, thereby limiting their artistic applicability. We present a method for nonlinear 3D face modeling using neural architectures that provides intuitive semantic control over both identity and expression by disentangling these dimensions from each other, essentially combining the benefits of both multi-linear face models and nonlinear deep face networks. The result is a powerful, semantically controllable, nonlinear, parametric face model. We demonstrate the value of our semantic deep face model with applications of 3D face synthesis, facial performance transfer, performance editing, and 2D landmark-based performance retargeting.

1. Introduction

Data-driven face models are very popular in computer vision and computer graphics, as they can aid in several important challenges like model-based 3D face tracking [12], facial performance retargeting [29], video based reenactment [32], and image editing [6]. These models are built from large databases of facial scans. Most commonly, linear face models are built, where the approximated face is expressed as a linear combination of the dataset shapes [7]. Extensions to multi-linear models [13, 33] also exist, which generate a tensor of different semantic dimensions (e.g. identity and expression). This ability to have semantic separation of attributes has several benefits, including for example constrained face fitting (e.g. fitting to an identity while constraining to the neutral expression, or fitting to an expression once the identity is known), performance animation (e.g. modifying only the expression space of the model), performance transfer or retargeting (modifying only the identity space of the model), etc. In general a model that provides semantic separation lends itself better to artistic control. The main problem with traditional models, however, is their linearity. The human face is highly nonlinear in its deformation, and it is well known that a simple blending of static expressions often results in unrealistic

motion. In severe cases, many combinations of the input expressions can lead to physically impossible face shapes (see Fig. 5). To summarize, linear models constrain the space of shapes to a manifold which on the one hand usually *cannot* represent all possible face shapes, and on the other hand *can easily* represent many non-face shapes.

As we shall see in more detail in Section 2, recent methods have begun to investigate nonlinear face models using neural networks [28, 1, 14, 20, 16, 24, 3], which can, to some degree, overcome the limitations of linear models. Unfortunately, some of these approaches have thus far sacrificed the human interpretable nature of multi-linear models, as one typically loses semantics when moving to a latent space learned by a deep network.

In this work, we aim to combine the benefits of multi-linear and neural face models by proposing a new architecture for *semantic* deep face models. Our goal is to retain the same semantic separation of identity and expression as with multi-linear models, but with deep variational networks that allow nonlinear expressiveness. To this end, we propose a network architecture that takes the neutral 3D geometry of a subject, together with a target expression, and learns to deform the subject's face into the desired expression. This is done in a way that fully disentangles the latent space of facial identities from the latent space of expressions. As opposed to existing deep methods [1, 14, 20, 16], the disentanglement is explicitly factored into our architecture and *not learned*. As a consequence, our method achieves perfect disentanglement between facial identity and expression in its *latent space*, while still encoding the correlation between identity and expression in *shape space*, i.e. the shape change induced by an expression differs as a function of the identity shape. Once trained (end to end), one can traverse the identity latent space to synthesize new 3D faces, and traverse the expression latent space to generate new 3D expressions, all with nonlinear behavior. Furthermore, since we condition the expression based on the popular representation of linear blendshape weights, the resulting network allows for semantic exploration of the expression space, which is also lacking in existing methods.

As face models that generate geometry alone have limited applicability, we further incorporate the appearance of the face into our architecture, in the form of a diffuse albedo texture map. An initial per-vertex color prediction that corresponds to the face geometry is transferred to the UV domain resulting in an low resolution texture map. We employ an image to image translation network [34] as a residual super-resolution network to transform the initial low resolution albedo to a resolution of 1024 x 1024.

We demonstrate the value of our semantic deep face model with several applications. The first is 3D face synthesis (see Section 4.2), where our method can generate a novel human face (geometry and texture) and the corresponding (nonlinear) expressions - a valuable tool for example in creating 3D characters in virtual environments. We also show that nonlinear 3D facial retargeting can be easily accomplished with our network, by swapping the identity latent code while keeping the per-frame expression codes fixed (see Section 4.3). Another application of our model is 3D face capture and retargeting from video sequences, by regressing to our expression latent space from 2D facial landmarks (see Section 4.3.2). Finally, in our supplementary video we demonstrate how our method allows an artist to edit a performance, e.g. add a smile/frown to certain keyframes of a captured facial performance. To summarize, we present a method for nonlinear 3D face modeling including both geometry and appearance, which allows semantic control by separating identity and expression in its latent space, while keeping them coupled in the decoded geometry space.

2. Related Work

Facial blendshapes [23] have been conventionally used as a standard tool by artists to navigate the space of the geometry of human faces. In addition to being human interpretable, blendshapes are extremely fast to evaluate, and enable artists to interactively sculpt a desired face. Blanz and Vetter [7] proposed a 3D morphable model of human faces, by using principal component analysis (PCA) to describe the variation in facial geometry and texture. Similarly, Vlasic et al. [33] proposed a multi-linear model based on tensor decomposition as a means of compressing a collection of facial identities in various expressions. However, in both morphable models and multi-linear tensors, the orthogonality comes at the cost of losing interpretability. In addition to linear global models, multi-scale approaches have been developed [27] [15], [9], with a focus on capturing and reconstructing local details and deformations. Building on top of the techniques mentioned above, several statistical models of the human face have also been built [13], [25], [8]. We refer to a comprehensive survey [10] of methods used in the statistical modelling of human faces.

Moving on to nonlinear geometry modelling, Tan et al. [31] proposed the use of a variational autoencoder (VAE) [22] to effectively compress and represent several categories of 3D shapes. They do so by describing the deformation of meshes in a local co-ordinate frame [26] and later reconstructing the positions of the mesh through a separate linear solve. In the context of human faces, Ranjan et al. [28] proposed the use of convolutional mesh autoencoders and graph convolutions as a means of expanding the expressiveness of face models. While they were able to achieve better reconstruction than linear models, disentangling facial identity and expression was not one of their objectives. Recent works [20, 2, 16, 14, 1] have begun to explore the disentanglement of facial identity and expression inside a neural network. The state of the art performance of these methods on standard datasets [13, 25] indicate the benefit of learning

disentangled representations with neural networks. However, these methods *learn* to disentangle latent identity and expression, while the disentanglement is factored by design into our architecture and is therefore more explicit. Additionally work such as [20, 2, 14, 1] do not jointly model facial geometry and appearance, while we do.

The more recent work of Li et. al [24] is closest in spirit to our work. Though our methods seem similar at the onset, there are a few important differences. The first is that though we decouple identity and expression in the network's latent space, our joint decoder can model identity specific expression deformations which [24] can not. Second, as we describe in Section 3.2, the manner in which we use dynamic facial performances for training readily makes our method applicable to retarget and reconstruct performance from videos, and addresses another limitation of [24]. Another interesting contribution in neural semantic face modelling is the work of Bailey et. al [3], where semantic control over expression is achieved through rig parameters instead of blendweights. However, since their method is rig specific, and doesn't model appearance, it unfortunately cannot be used for several of the applications demonstrated in this work.

In this work, we extend the state of the art in non-linear semantic face models, by proposing a novel neural architecture that explicitly disentangles facial identity and expression in its latent space, while retaining identity-expression correlation in geometry and appearance space. Through the use of blendweights, our method provides intuitive control over the generated expressions, retaining the benefits of traditional multi-linear models, with increased expressiveness, and lends itself to applications in 3D face synthesis, 2D and 3D retargeting, and performance manipulation.

3. Methodology

We now present our method, starting with an overview (Section 3.1), our data acquisition and processing steps (Section 3.2), a description of the main architecture for semantically generating face geometry and low resolution appearance (Section 3.3), our appearance super-resolution approach (Section 3.4), and details on training (Section 3.5).

3.1. Concept Overview

In this work, we assume that we are given access to a 3D face database consisting of several subjects in a fixed set of expressions, where the meshes are assumed to be in full vertex correspondence, similar to the datasets that traditional face models are built from. Our method can optionally also take appearance data in the form of per-vertex color information, corresponding to each expression. In addition to the static expressions, access to registered dynamic performances of subjects can also be used whenever available (although dynamic data is not mandatory). We address how such a database can be built in Section 3.2. We propose a novel neural approach to human face modelling consisting of a pair of two variational auto-encoders (VAE), which use such a database to build a latent space where facial identity and expression are guaranteed to be disentangled by design, while at the same time allowing a user to navigate this latent space with interpretable blendweights corresponding to semantic expressions.

Given the neutral geometry and albedo of a subject, and a target blendweight vector, our collection of networks learn to deform the subject's neutral into the desired captured expression, and also generate the corresponding per-vertex albedo. In the process of doing so, an identity VAE projects the subject's face onto a latent space of facial identities while an expression VAE projects the target blendweight vector into a latent expression space. By combining the information from the identity and expression embeddings, a joint decoder learns the nonlinearity of facial deformation to produce per-vertex displacements that deform the given neutral into the desired expression, along with non-linear albedo displacements that represent a corresponding expression-specific albedo. Our VAE learns the high-level correlation between the facial geometry and albedo. The per-vertex albedos are sampled as texture images in the UV domain at relatively low resolution, and are then upsampled with a variant of the *Pix2PixHD* architecture [34] in order to generate a high-resolution detailed facial textures.

3.2. Data Acquisition and Processing

Before we describe our algorithm in detail, a fundamental requirement of our method is a registered 3D face database of different subjects performing a variety of facial expressions. Since most existing 3D databases of human faces [25, 13, 8] are limited in their geometric resolution, and lack either variations in the identities of subjects [25, 13] or do not contain sufficient examples of the same subject performing different expressions, we capture and build our own 3D facial database. In a passively lit, multi-camera setup, we capture 224 subjects of different ethnicities, genders, age groups, and BMI. Subjects were carefully chosen such that each of the sampled distributions are as uniformly represented as practically possible. Each of the 224 subjects was captured performing a pre-defined set of 24 facial expressions, including the neutral expression. In addition to capturing the static expressions of 224 subjects, we also captured a dynamic speech sequence and a facial workout sequence for a subset of 17 subjects. The captured images of subjects in various expressions were reconstructed using the method of Beeler et. al [4]. A template mesh consisting of 49,000 vertices was semi-automatically registered to the reconstructions of each subject individually, and a 1024x1024 albedo texture map was generated by dividing out the diffuse shading given a measured light

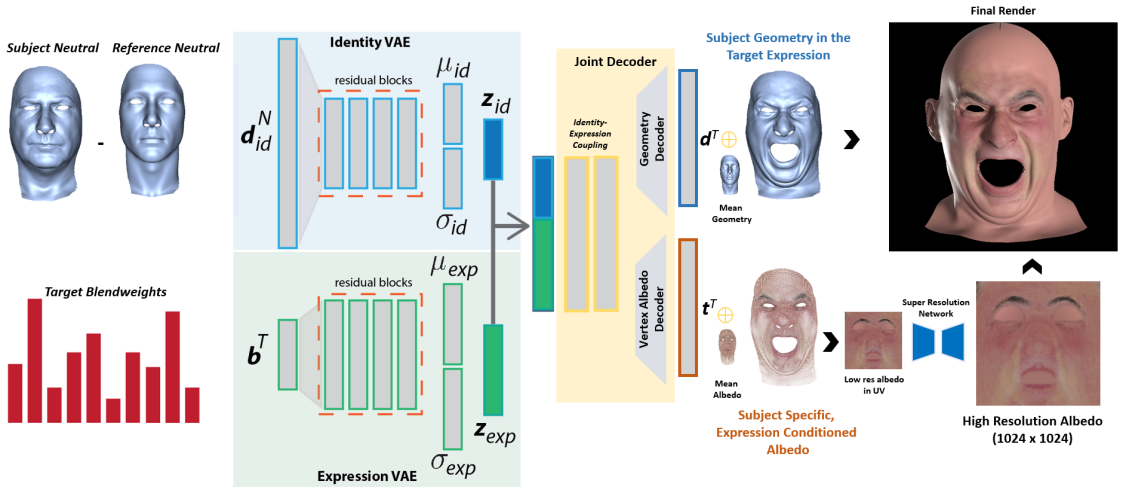


Figure 2: Our network architecture for semantic deep faces. We disentangle identity and expression through separate VAEs, which are trained end-to-end with a joint decoder given a subject's neutral and target expression with known target blendweights. The joint decoder outputs the deformed geometry and corresponding per-vertex albedo. The low resolution albedo, after being transferred to the UV domain, is passed through a super resolution network to result in the final albedo. The synthesized geometry and albedo can be used to render realistic digital human faces.

probe. As a result of this, we end up with a total of 5,376 meshes and textures (224 subjects x 24 expressions) that are in full correspondence with one another. We further stabilize the expression to remove any rigid head motion [5] and align all of them to the same canonical space. For training the albedo model, we sample the per-vertex albedo color and store the RGB information with each vertex, forming a 6-dimensional vector (XYZRGB). For the subjects for whom dynamic performances were captured, we start from their registered static meshes and build a subject specific anatomical local face model [35]. This subject specific model is then used to track the dynamic performance of the subject. For the 17 subjects we recorded, we reconstructed and tracked a total of 7,300 frames. Next, we associate blendweight vectors to each registered mesh. For the static shapes, since each mesh corresponds to a unique, pre-defined expression, the blendweight vectors are one-hot encoded vectors corresponding to the captured expression. This results in the assignment of a 24 dimensional blendweight vector $\mathbf{b} \in \mathcal{R}^{24}$ to each shape. However, the assignment of blendweight vectors for a dynamic shape is not straightforward as the subject may have performed an expression that could only be explained by a combination of the individual shapes. Therefore, we fit a weighted combination of the 24 registered shapes of the subject in a least squares sense to the tracked performance. This gives us optimal blendweights for each frame in the performance. As we will show later (Fig. 5 (c)), the linear blendshape fit is only a crude approximation of the real shape. We therefore discard the linear shape estimate (keeping only the optimized blend weights) and use the captured shape as ground truth to train our decoder. This way, we can leverage both static and dynamic data for training. A conceptual overview of our architecture is shown in Fig. 2.

3.3. A Variational Multi-Nonlinear Face Model

From the database described in Section 3.2, we compute the mean of all subjects in the neutral expression and call this shape the *reference mesh* R. We then subtract R from the original shapes, providing us with per-vertex displacements for each identity in the neutral expression. We identically pre-process the per-vertex albedo by subtracting the mean from each of the training samples. We will describe the model now in the context of one subject, where subscripts id and exp represent the identity and expression components of the subject, respectively, and superscripts N and T correspond specifically to neutral and target expression shapes.

The mean-subtracted neutral displacements $\mathbf{d}_{id}^N$ are fed as the input to an identity VAE. We use displacements rather than other representations like the linear rotation invariant (LRI) coordinates [26] as used by Tan et. al [31] since our input shapes are carefully rigidly stabilized. Our identity encoder E_{id} is a fully connected network consisting of residual blocks that compress the input displacements into a mean μ_{id} and standard deviation σ_{id}.

$$\mu_{id}, \sigma_{id} \leftarrow E_{id}(\mathbf{d}_{id}^N). \tag{1}$$

At training time, the predicted mean and standard deviation vectors are used to sample from a normal distribution using the re-parametrization method of Kingma et. al. [22] to produce a n_{id}-dimensional *identity code* $\mathbf{z}_{id}$.

$$\mathbf{z}_{id} \sim \mathcal{N}(\mu_{id}, \sigma_{id}). \tag{2}$$

The output of each fully connected layer except the ones predicting the mean and the standard deviation are activated with a leaky ReLU function. The identity encoder only ever sees the displacements of different subjects in the neutral expression, crucial for the decoder to explicitly decouple identity and expression.

In parallel, a second expression VAE, E_{exp}, takes a blendweight vector $\mathbf{b}^T$ corresponding to target expression T as its input and compresses or expands it into a variational latent space $\mathbf{z}_{exp}$ of n_{exp} dimensions. Similar to the identity encoder, the expression VAE is also a fully connected network with residual blocks and leaky ReLU activations. The expression VAE also outputs a mean and standard deviation vector that are fused into the *expression code* $\mathbf{z}_{exp}$.

$$\mu_{exp}, \sigma_{exp} \leftarrow E_{exp}(\mathbf{b}^T) \tag{3}$$

$$\mathbf{z}_{exp} \sim \mathcal{N}(\mu_{exp}, \sigma_{exp}). \tag{4}$$

Our choice to use blendweights to condition the decoder is motivated by two reasons. The first is that blendweights provide a semantic point of entry into the network and can therefore be manipulated at test time by an artist. Second, one of our objectives is to force the network to disentangle the notion of facial identity and expression. Blendweights are a meaningful representation to learn this disentanglement as they contain no notion of identity and are purely descriptive of expression. The identity and expression codes are concatenated into a vector of dimension n_{id} + n_{exp} and fed to a decoder D that learns to correlate the identity and expression spaces and eventually reconstructs the given identity in the desired expression with corresponding per-vertex albedo estimate. The decoder is a fully connected network that outputs vertex displacements $\mathbf{d}^T$ with respect to the reference mesh R, and albedo displacements $\mathbf{t}^T$ as

$$[\mathbf{d}^T, \mathbf{t}^T] \leftarrow D(\mathbf{z}_{id}, \mathbf{z}_{exp}). \tag{5}$$

Disentanglement by Design The joint decoder takes the two variational codes produced independently by the two VAEs to reconstruct the input subject in the desired expression. Since the two latent codes are fully disentangled, the decoder must learn to correlate identity and expression codes to reconstruct the training shapes. This combination of a disentangled latent space and correlated geometry space enables to capture identity specific deformations (in both shape and albedo) for the same semantic expression, as shown in Fig. 3.

We use four residual layers in both E_{id} and E_{exp}, where the dimensions of the layers are fixed to n_{id} and n_{exp}, respectively. Following our experiments outlined in Section 4, we set $n_{id} = 32$ and $n_{exp} = 256$ for all results. We resorted to the use of a VAE as opposed to a generative model to avoid running into mode collapses and to compensate for the lack of extensive training data. Our disentanglement framework is otherwise generic and could readily benefit from the use of graph convolutions [28] and other neural concepts that focus on reconstruction accuracy. In other words, the novelty of our method primarily stems from our ability to semantically control a powerful nonlinear network while ensuring that it's internal representations fully disentangle facial identity and expression.

3.4. Appearance Super-Resolution

The predicted per-vertex albedo displacements $\mathbf{t}^T$ are added to the mean albedo and transferred to the UV domain. As seen in Fig. 2, the resulting texture map contains coarse information, such as the global structure of the face (the position of the eyes, mouth etc.), expression dependent effects (blood flow), as well as identity cues (ethnicity, gender etc.). What is missing are the fine details that contribute to the photo-realistic appearance of the original high resolution albedo. Our goal is to regenerate these missing details conditioned by the low resolution albedo, upscaled to the target resolution. We reformulate this super-resolution task as a residual image-to-image translation problem [19], trained on the captured high resolution albedo texture maps. The low resolution albedo is upscaled using bilinear interpolation to the target resolution (1024 x 1024). The upscaled albedo A^{Up} is then fed to a generator G_{Res} [34] that outputs a residual image A^{Res}, which is combined with with A^{Up} to produce the final texture $A^{'}$. The discriminators that provide adversarial supervision to the generator are multiple Markovian patch-based discriminators D_p, each of which operates at a different scale p of the input. We do not use any normalization layers in both the generator and the discriminators.

3.5. Training

Geometry VAEs: The identity and expression VAEs, along with the joint decoder, are trained end-to-end in a fully supervised manner using both static and dynamic performances. We penalize the reconstructed geometry with a L1 loss, and the identity and expression latent spaces are constrained using the KL divergence. Training takes around 4 hours on single Nvidia 1080 Ti GPU. We initialize both encoders and the decoder following Glorot et. al [17], and use the ADAM optimizer [21] with a learning rate of 5e-4.

Albedo Super-Resolution: The residual generator G_{Res} is trained akin to the generator in [34], using both ground truth and adversarial supervision. For ground truth supervision with the captured high resolution albedo A^{GT}, we use an L1 loss (L_1) and the VGG-19 [30] perceptual loss L_{VGG}. We train each discriminator D_p using the WGAN-GP loss as proposed by Gulrajani et. al [18]. We use a learning rate of 1e-4 and optimize the generator and discriminators using

the ADAM optimizer [21]. We refer to our supplementary material for additional details on the network architecture and loss formulations.

4. Results and Discussion

Our goal is to produce a semantically controllable, nonlinear, parametric face model. In this section we inspect the disentangled latent spaces for identity and expression, and show how the nonlinear representation is more powerful than traditional (multi-)linear models, while providing the same semantic control.

4.1. Quantitative Evaluation on Facewarehouse

The Facewarehouse dataset [13] contains meshes of 150 identities in 47 different expressions, where each mesh contains 11,518 vertices. Since the meshes in Facewarehouse do not have an associated texture map, we train only the geometry decoder (Fig. 2) for this experiment. Similar to Jiang et. al [20], we train our model on an augmented set of the first 140 identities and their expressions, and test on the 10 remaining identities.

The table in Fig. 4 (left) compares our reconstruction accuracy on the Facewarehouse dataset to existing state of the art in 3D face modelling. To enable a fair comparison to existing work, we also fix the total dimensionality of our latent spaces to 75 dimensions like other works. See the supplementary material for qualitative results on the Facewarehouse dataset.

4.2. Disentangled Latent Spaces

Our disentangled representation allows for smooth control over both identity and expression independently.

4.2.1 Identity Latent Space

Varying the identity code while keeping the expression code fixed will produce different identities with the same expression. Fig. 3 (a) (top 2 rows) shows the result of random samples drawn from the identity latent space, also rendered with the resulting upsampled albedo. The choice of a variational autoencoder to represent the identity space allows to smoothly morph between different subjects by (linearly) interpolating their identity codes. As Fig. 3 (b) shows, the degree of nonlinearity reflected in the output shapes varies as a function of the dimensionality of the latent space, where a lower dimensionality will force higher nonlinearity. Notice how interpolating between two identities appears to pass through other identities for lower dimensional identity spaces. While a lower dimensional latent space reduces the reconstruction accuracy (see Fig. 4) due to the higher compression, our representational power is still significantly higher than a linear model (PCA). Increasing dimensions diminishes this advantage due to the relatively low number of training samples.

4.2.2 Expression Latent Space

While it would be an option to directly sample the expression latent space analogous to the identity latent space, this would not allow for semantically meaningful control. For human animators it is critical to provide an intuitive control structure to animate the face, referred to as *rig*. The most well-known rigging concept for facial animation are blendshapes, which are extremely intuitive as they allow the animator to dial in a certain amount of a given expression. These can then be superimposed to provide the final shape. In our system, the exposed expression controls are provided in exactly the same way, via a vector of blendweights that encode the intensity of the individual shapes to be dialed in. Due to the disentangled nature of identity and expression spaces, it is possible to synthesize any desired expression as shown in the bottom part of Fig. 3 (a) for a given identity. Here we provide one-hot blendweight vectors to the network and generate the complete set of blendshapes. As such, the proposed model can be readily adopted by animators. Corresponding high resolution albedo textures for the synthesized expressions are also produced by our method, as illustrated in the expression interpolation example in Fig. 3 (c). In addition to providing an interface akin to blendshapes, our method has quite some advantages over a linear blendshape basis. Fig. 5 (a) shows that our model is much more robust when extrapolating along an expression dimension beyond [0,1], unlike the linear model, which leads to exaggerated and unusable shapes, especially towards the negative direction. Furthermore, linearly varying the weight within [0,1] provides a nonlinear effect on the generated shape, as demonstrated on the smile expression, where the generated smile starts off as a closed mouth smile up until $\sim$0.6, and then opens up, which feels more natural than the monotonous interpolation of the linear model. This nonlinearity is especially important when superimposing expressions (Fig. 5 (b)). For a linear model, the latter only makes sense for a few combinations of expressions, and hence blendshape editing often yields undesired shapes quickly, especially for novice users, whereas the proposed method is more robust in such cases. As expected, our nonlinear model has higher expressive power than its linear counterpart (Fig. 5 (c)) when fitting to a ground-truth reconstructed performance, the linear model incurs a larger reconstruction error for the same blend vector dimensionality. Using the fitted linear blendweights as input to our network, our method achieves much lower errors, close to the optimal expression the model can produce, found in this case by optimizing in the expression latent space.

4.3. Facial Performance Retargeting

Our method also lends itself to facial performance transfer using blendweights or 2D landmarks.

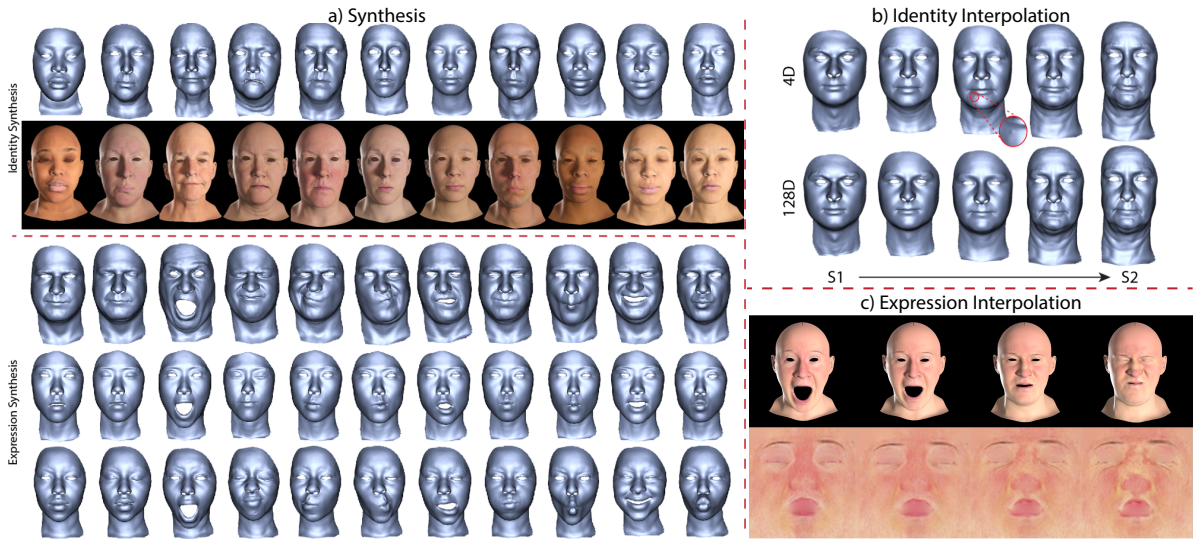


Figure 3: **a) Face synthesis results** Here we show a set of identities synthesized by sampling the identity latent code (top 2 rows, with completed 3D head geometry rendered with our synthesized albedo, as well as a subset of expressions for three different identities produced by sampling the expression latent space (bottom rows). **b) Identity interpolation** between two subjects in latent identity spaces of different dimensions (top row 4D, bottom row 128D). The lower dimensional space passes through other identities as we interpolate (notice the mole on the chin of the center subject which is not present in either the start or end identity). **c) Expression interpolation** *Top:* Here we see the change in geometry as we interpolate between two expressions for a synthesized subject while keeping the identity code fixed. *Bottom:* We see the corresponding albedo as generated by our networks. Notice how our method can capture expression specific changes in facial appearance; especially around the nose for this example.

a) Facewarehouse evaluations

Method	E_{avd} (mm)	
	Mean	Median
Bilinear [13]	0.993	0.998
FLAME [25]	0.882	0.905
MeshAE [28]	0.825	0.811
Jiang et.al [20]	0.472	0.381
Ours	**0.376**	**0.351**

b) Comparisons with PCA on novel identities

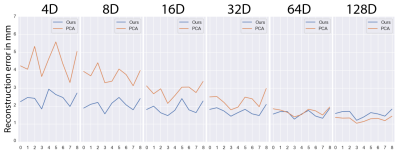


Figure 4: *Left*: In this table, we present quantitative comparisons of our method against state of the art on the facewarehouse dataset. *Right*: Reconstruction error on 9 validation shapes from our in house dataset which are not part of the model. The nonlinear model has lower error than the linear one for low dimensional latent spaces. At 64 dimensions the two models are comparable, and at 128 dimensions the linear model is actually superior, as there are insufficient samples to train such a high-dimensional space.

4.3.1 Blendweight Retargeting

Retargeting performances by transferring the semantic blendweights from one character to another is a common approach in facial animation. The same paradigm can be used with our nonlinear face model, by first determining the identity code of the target actor using the identity VAE (given the target neutral expression), and then injecting the per-frame blendweights to the expression VAE. Fig. 6 illustrates this procedure, transferring the expression weights obtained from a performance onto a novel identity.

4.3.2 2D Landmark-Based Capture and Retargetting

Another interesting scenario is facial performance capture and retargeting based on 2D facial landmarks in videos. Here we show an extension of our architecture that allows an interface to the latent expression code via 2D landmarks. Given a subset of our facial database where frontal face imagery is available, we detect a typical landmark set [11] and perform a normalization procedure to factor out image translation and scale (based on the inter-ocular distance). The normalized landmarks are then stacked into a vector, and fed to a network that aims to map the landmarks to the corresponding expression code $\mathbf{z}_{exp}$. We illustrate this landmark architecture in Fig. 7 (left). The network is trained with ground truth blendweights which allows supervision on the expression code, given the pre-trained expression VAE, and we include the resulting geometry in the loss function using the pre-trained decoder. The result is a means to generate expressions based on 2D landmarks, which allows further applications of our deep face model including landmark-based performance capture (Fig. 7 right - center row) and retargeting to a new identity (bottom row).

4.4. Limitations and Future Work

While the proposed expression encoding is more robust to random blendweight combinations than linear models, it

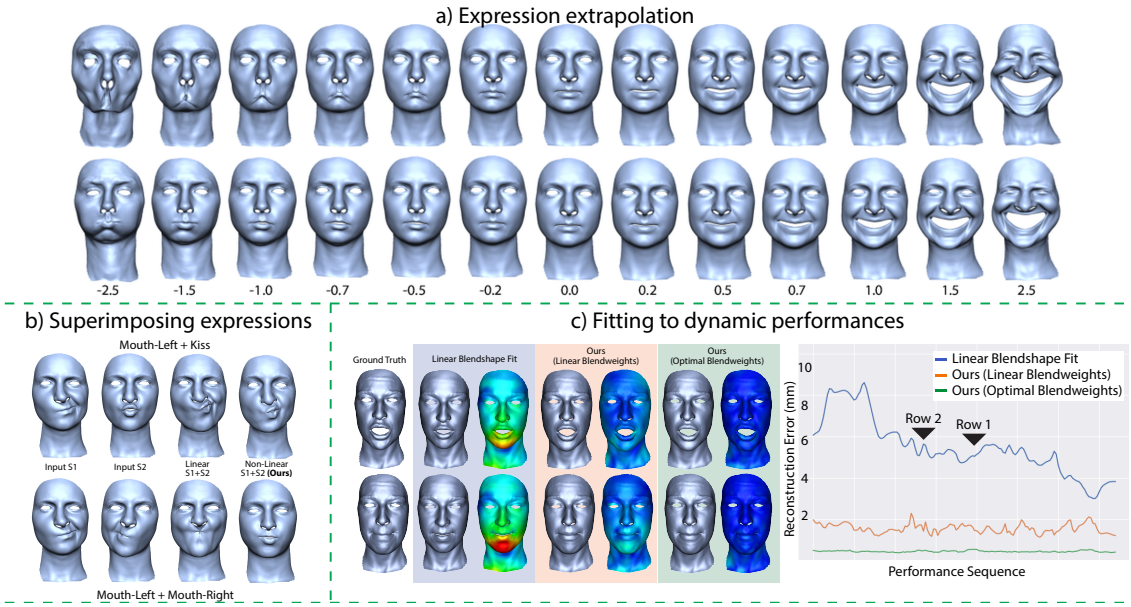


Figure 5: **(a) Expression extrapolation** Our nonlinear model (bottom row) allows to semantically control the intensity of an expression akin to a linear blendshape model (top row). However, the nonlinear model extrapolates better, producing plausible shapes within [-1,1] and degrading gracefully beyond, unlike the linear model. Furthermore, the expression changes on a nonlinear trajectory, e.g. causing the smile to start as a closed smile (up to 0.6) and then open up in a natural way compared to the steady increase in the linear model. **(b) Superimposing expressions** The linear model can superimpose only non-conflicting expressions, such as mouth-left and kiss (top), but generates poor results for many shape combinations, such as mouth-left with mouth-right (bottom). Our nonlinear model produces more plausible shapes in such cases. **(c) Fitting to dynamic performances** Comparing the reconstruction residual of the linear blendshape fit with that of our model shows that our model has higher representational power. Heatmap encodes errors from 0 mm (blue) to 15 mm (red).



Figure 6: Retargeting a facial performance from one actor to another by fixing the blendweights and changing the identity code results in a natural-looking transfer.

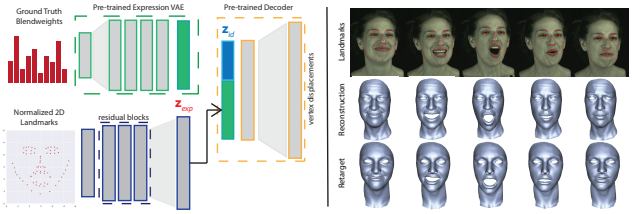


Figure 7: *Left:* Our landmark architecture allows to train a mapping from 2D face landmarks to the expression latent space. *Right:* Landmark-based generation of a 3D performance (center row) from a 2D video of the person (top row) and retargeting to any other identity (bottom row).

is however not guaranteed to produce meaningful shapes for any given blendweight vector. It would be very valuable to have a representation that maps the unit hypercube to the physically meaningful expression manifold in order to allow random sampling that provides valid shapes spanning the complete expression space. Even though we incorporate dynamic performances, we do not encode the temporal information, which would allow to synthesize temporal behaviour, such as nonlinear transitioning between expressions. Lastly, we feel the proposed approach is not limited to faces but could provide value in other fields, for example general character rigging.

5. Conclusion

We propose semantic deep face models—novel neural architectures for 3D faces that separate facial identity and expression akin to traditional multi-linear models, but with added nonlinear expressiveness, and the ability to model identity specific deformations. We believe that our method for disentangling identity from expression provides a valuable, semantically controllable, nonlinear, parametric face model that can be used in several applications in computer vision and computer graphics.

References

[1] V. F. Abrevaya, A. Boukhayma, S. Wuhrer, and E. Boyer. A generative 3d facial model by adversarial training. *CoRR*, abs/1902.03619, 2019. 2, 3

[2] T. M. Bagautdinov, C. Wu, J. M. Saragih, P. Fua, and Y. Sheikh. Modeling facial geometry using compositional vaes. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3877–3886, 2018. 2, 3

[3] S. W. Bailey, D. Omens, P. Dilorenzo, and J. F. O'Brien. Fast and deep facial deformations. *ACM Transactions on Graphics*, 39(4):94:1–15, Aug. 2020. Presented at SIGGRAPH 2020, Washington D.C. 2, 3

[4] T. Beeler, B. Bickel, P. Beardsley, B. Sumner, and M. Gross. High-quality single-shot capture of facial geometry. *ACM Trans. Graph.*, 29(4):40:1–40:9, July 2010. 3

[5] T. Beeler and D. Bradley. Rigid stabilization of facial expressions. *ACM Transactions on Graphics (TOG)*, 33(4):44, 2014. 4

[6] V. Blanz, K. Scherbaum, T. Vetter, and H.-P. Seidel. Exchanging Faces in Images. *Computer Graphics Forum*, 2004. 1

[7] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, pages 187–194, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. 1, 2

[8] J. Booth, A. Roussos, S. Zafeiriou, A. Ponniah, and D. Dunaway. A 3d morphable model learnt from 10,000 faces. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5543–5552, 2016. 2, 3

[9] A. Brunton, T. Bolkart, and S. Wuhrer. Multilinear wavelets: A statistical shape space for human faces. *CoRR*, abs/1401.2818, 2014. 2

[10] A. Brunton, A. Salazar, T. Bolkart, and S. Wuhrer. Comparative analysis of statistical shape spaces. *CoRR*, abs/1209.6491, 2012. 2

[11] A. Bulat and G. Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230, 000 3d facial landmarks). *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 1021–1030, 2017. 7

[12] C. Cao, Y. Weng, S. Lin, and K. Zhou. 3d shape regression for real-time facial animation. *ACM Trans. Graph.*, 32(4):41:1–41:10, 2013. 1

[13] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):413–425, March 2014. 1, 2, 3, 6

[14] V. Fernández Abrevaya, S. Wuhrer, and E. Boyer. Multilinear autoencoder for 3d face model learning. In *Applications of Computer Vision (WACV), 2018 IEEE Winter Conference on*, 2018. 2, 3

[15] C. Ferrari, G. Lisanti, S. Berretti, and A. Del Bimbo. Dictionary learning based 3d morphable model construction for face recognition with varying expression and pose. In *Proceedings of the 2015 International Conference on 3D Vision*, 3DV '15, pages 509–517, Washington, DC, USA, 2015. IEEE Computer Society. 2

[16] B. Gecer, A. Lattas, S. Ploumpis, J. Deng, A. Papaioannou, S. Moschoglou, and S. Zafeiriou. Synthesizing coupled 3d face modalities by trunk-branch generative adversarial networks. *ArXiv*, abs/1909.02215, 2019. 2

[17] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 9:249–256, 13–15 May 2010. 5

[18] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. *CoRR*, abs/1704.00028, 2017. 5

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *arxiv*, 2016. 5

[20] Z.-H. Jiang, Q. Wu, K. Chen, and J. Zhang. Disentangled representation learning for 3d face shape. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 3, 6

[21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 5, 6

[22] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013. 2, 4

[23] J. P. Lewis, K. Anjyo, T. Rhee, M. Zhang, F. Pighin, and Z. Deng. Practice and Theory of Blendshape Facial Models. In S. Lefebvre and M. Spagnuolo, editors, *Eurographics 2014 - State of the Art Reports*. The Eurographics Association, 2014. 2

[24] R. Li, K. Bladin, Y. Zhao, C. Chinara, O. Ingraham, P. Xiang, X. Ren, P. Prasad, B. Kishore, J. Xing, and H. Li. Learning formation of physically-based face attributes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2, 3

[25] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), 2017. 2, 3

[26] Y. Lipman, O. Sorkine, D. Levin, and D. Cohen-Or. Linear rotation-invariant coordinates for meshes. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, 24(3):479–487, 2005. 2, 4

[27] T. Neumann, K. Varanasi, S. Wenger, M. Wacker, M. Magnor, and C. Theobalt. Sparse localized deformation components. *ACM Trans. Graph.*, 32(6):179:1–179:10, Nov. 2013. 2

[28] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black. Generating 3D faces using convolutional mesh autoencoders. In *European Conference on Computer Vision (ECCV)*, volume Lecture Notes in Computer Science, vol 11207, pages 725–741. Springer, Cham, Sept. 2018. 2, 5

[29] R. B. i. Ribera, E. Zell, J. P. Lewis, J. Noh, and M. Botsch. Facial retargeting with automatic range of motion alignment. *ACM Trans. Graph.*, 36(4):154:1–154:12, 2017. 1

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015. 5

[31] Q. Tan, L. Gao, Y. K. Lai, and S. Xia. Variational Autoencoders for Deforming 3D Mesh Models. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2018. 2, 4

[32] J. Thies, M. Zollhöfer, M. Stamminger, C. Theobalt, and M. Nießner. Face2Face: Real-time Face Capture and Reenactment of RGB Videos. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2016. 1

[33] D. Vlasic, M. Brand, H. Pfister, and J. Popović. Face transfer with multilinear models. *ACM Trans. Graph.*, 24(3):426–433, July 2005. 1, 2

[34] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2, 3, 5

[35] C. Wu, D. Bradley, M. Gross, and T. Beeler. An anatomically-constrained local deformation model for monocular face capture. *ACM Trans. Graph.*, 35(4):115:1–115:12, July 2016. 4