# Semantify: Simplifying the Control of 3D Morphable Models using CLIP

Omer Gralnik
Reichman University

Guy Gafni
Technical University of Munich

Ariel Shamir
Reichman University

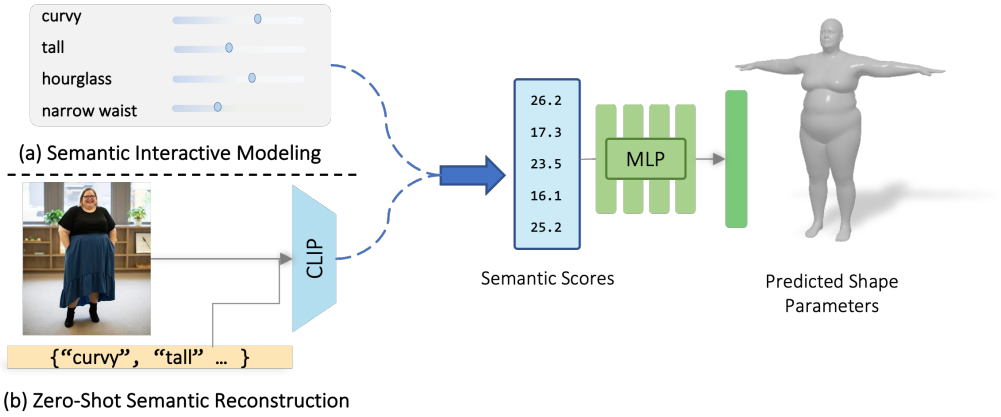


Figure 1: Semantify offers a method to create and edit a 3D parametric model using semantically meaningful descriptors. Semantify is based on a self-supervised method that utilizes the semantic power of CLIP language-vision model to build a mapping between semantic descriptors to 3DMM model coefficients. This can be used in an interactive application defining a slider for each descriptor (a), or to fit a model to an image in a zero shot manner by feeding the image into CLIP and obtaining a vector of semantic scores that can be mapped to shape parameters (b).

## Abstract

*We present Semantify: a self-supervised method that utilizes the semantic power of CLIP language-vision foundation model [32] to simplify the control of 3D morphable models. Given a parametric model, training data is created by randomly sampling the model's parameters, creating various shapes and rendering them. The similarity between the output images and a set of word descriptors is calculated in CLIP's latent space. Our key idea is first to choose a small set of semantically meaningful and disentangled descriptors that characterize the 3DMM, and then learn a non-linear mapping from scores across this set to the parametric coefficients of the given 3DMM. The non-linear mapping is defined by training a neural network without a human-in-the-loop. We present results on numerous 3DMMs: body shape models, face shape and expression models, as well as animal shapes. We demonstrate how our method defines a simple slider interface for intuitive modeling, and show how the mapping can be used to instantly fit a 3D parametric body shape to in-the-wild images. See our project page at https://omergral.github.io/Semantify/*

## 1. Introduction

3D modeling techniques have evolved tremendously over the last few years. Such techniques are used in countless industries including the burgeoning AR/VR industry, fashion design, game development, film, and many others. However, designing a high-quality 3D model is not a simple task for most people and might require a well-trained 3D artist. Even when using more recent parametric morphable models (3DMM) [24, 31, 45, 3, 42] it is still hard for humans to understand how to choose the correct set of parameters, e.g. to achieve a specific human body shape or facial expression. It is also difficult to find what are the limits of the given parametric model in terms of coverage and expressiveness. The reason is that in most cases, the provided set of parameters is not interpretable, as they are commonly calculated using automatic optimization mechanisms followed by dimensionality reduction using PCA. Thus, they carry no clear semantic meaning.

To simplify the use of 3DMMs and allow for natural interactive human modeling, a key research question is – how to insert semantics into 3DMM control? Previous approaches [34] relied mostly on human intelligence and la-

beling, which is often time consuming and expensive. In addition, previous methods create a large number of semantic control descriptors that are often correlated and entangled. This means it is difficult to anticipate the effect of each descriptor as changing a value in one may modify others, resulting in difficulties to control the model.

In this paper, instead of using human labeling, we rely on the remarkable abilities of huge foundation models that combine natural language and visual understanding. We present a *self-supervised* method that utilizes the semantic power of CLIP [32] to define a method to control 3DMMs that carries two main advantages. First, it is more natural for humans to use as it allows modeling using a *small* number of *semantically meaningful* descriptors, that cover the space of deformations but are *disentangled*. Second, it covers even extreme examples in the shape/pose space of parametric models as it utilizes CLIP's pre-training on a large number of images. The main idea is first to use CLIP to select a small subset of semantically meaningful and disentangled descriptors, and then learn a non-linear mapping of this set to the coefficients of the given 3DMM.

Given a parametric 3DMM, we sample its parameter space to create a dataset containing a variety of 3D mesh shapes. We then render each mesh from different camera views to create a diverse set of images corresponding to the parameter samples (see Figure 2). Next, we gather a set of semantically descriptive textual terms related to the parametric 3D model, which we call *descriptors*. We encode both the images (using CLIP's image encoder) and the descriptors (using CLIP's text encoder) into CLIP's latent space and compare them. This defines a vector of similarity scores between the input vector of 3DMM coefficients (of each image sample) and each corresponding semantic descriptor. Next, we define a selection scheme to choose a small number of descriptors that are de-correlated to control the model. Lastly, we train a neural network to learn a mapping between the vector of similarity scores to the vector of 3DMM parameters.

We demonstrate how the learned mapping can be used to define an interface to control a 3DMM in a way that is simple and effective using a small set of semantically meaningful sliders (see Figure 1 and 4). Such sliders are easy to employ for designing high-quality 3D models and cover the shape space well. We demonstrate this for four parametric models: human face's shape and expression (FLAME [24]), human body shapes (SMPL [25] and SMPL-X [31]), and even animals (SMAL [45]). We also show how the mapping can be used to instantly fit a 3D parametric body shape to an input image that works well "in the wild" even in extreme poses and body shapes.

Our main contribution is a novel, self-supervised method for defining a small set of semantic descriptors to control a parametric model more naturally, by learning a mapping from a semantic space to a parametric representation without human-in-the-loop. We demonstrate the effectiveness of our approach on several parametric models and utilize it to define a simple interface for modeling and to instantly fit a 3D model to images in the wild. We will release our code for future research.

## 2. Related Work

Our work encompasses a wide range of disciplines, including 3D parametric modeling, language-vision models, and zero-shot reconstruction.

**3D Morphable Models.** Much research has gone into improving 3D modeling of humans bodies, faces and other object classes, such as animals [11, 4, 25, 31]. 3D Morphable Models (3DMM) are a powerful tool to parameterize the variation in geometry of objects belonging to a certain class. Dating back to 1999, Blantz et al. [4] proposed 3DMMs to capture the variation of human faces and applied dimensionality reduction using Principal Component Analysis (PCA).

In contrast to a 3D artist freely controlling the vertices of a mesh, using each of these PCA axes to deform the vertices induces a strong statistical prior on the resulting geometry. The resulting deformations are global in nature, strongly constraining the geometry to the manifold of statistically plausible meshes, as the per-axis coefficients correlate with their likelihood. While initially proposed for human faces, statistical 3DMMs have been developed for human body models [3, 25, 40, 31, 42, 29], facial models [24, 37, 41, 43], and even 3D animal models [45, 44].

**Semantic Models.** PCA-based 3DMMs are efficiently and easily constructed, and admit an orthogonal and importance-ordered basis for the deformation. One disadvantage of these models is their lack of interpretability for humans. The axes of maximal variation in geometry, which form the deformation basis for each vertex, do not necessarily correspond to any semantically meaningful geometric change (e.g., raising an eyebrow), but instead, each axis induces a rather global change that may introduce correlations or effects that are unwanted to a human modeler, making it hard to obtain the desired shape or expression.

Some 3DMMs [17, 14] alleviate this issue by using deformation bases that are hand-crafted by artists, often referred to as *blendshapes*. While this alleviates the interpretability issue, it loses the data-driven and explicit statistical prior, requires manual work, and does not guarantee the expressiveness of the resulting model.

First to address the semantic issue of PCA-based 3DMMs in the human body domain were Seo et al. [33] which used metrics such as hip-to-waist ratio, body fat percentage, and height to regress 3DMM shape parame-

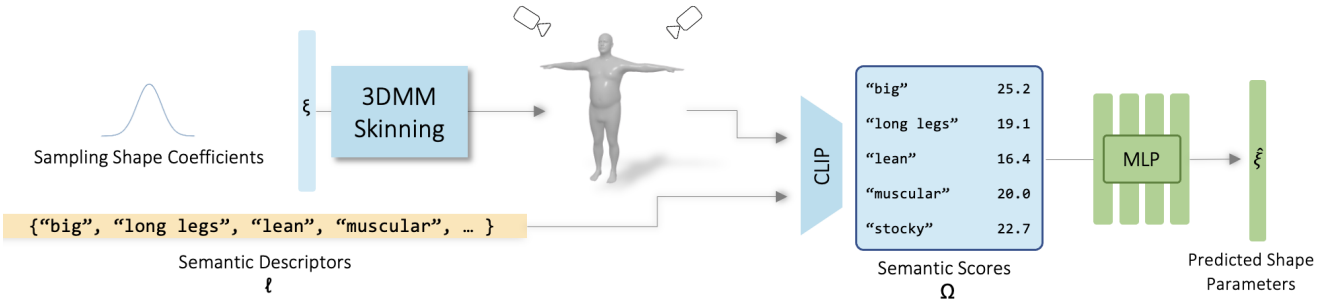


Figure 2: Learning a mapping from Semantic to Parametric space. (a) Given a coefficient vector $\vec{\xi}$ we create the 3DMM mesh. The mesh is rendered from several views. Each Rendered image $I'$ is passed into CLIP along with a set of semantic descriptors $\vec{\ell}$. The difference between each descriptor and the image in CLIP latent space is calculated and stored in the corresponding entry of the similarity vector $\vec{\Omega}$. (b) Using a large set of such random pairs of $(\vec{\Omega}, \vec{\xi})$, we train a network to learn the mapping from semantic space to parametric space.

ters. Hill et al.[15] suggested examining the relationship between body shapes and description, by linking two similarity spaces - one created from body descriptions and the other from full-body laser scans. BodyTalk [34] followed suit and approached this problem using the "wisdom of the crowd". In their work, they captured 256 male and female body shapes and represented each shape with 30 body descriptor words (e.g. curvy, fit, heavyset, round-apple, etc.). They hired labelers to discretely rate the bodies with respect to these descriptors and designed a regression model to learn the relationship between the discrete ratings and the parameters of the corresponding body shape. In a more recent publication, SHAPY [7] bridges this gap by curating datasets of images with corresponding body measurements (e.g., from model agencies) followed by human ratings respective to a set of linguistic shape attributes. They train models to predict SMPLX shape parameters from attributes and/or body measurements, and vice-versa.

**Language-Guided 3D Modeling.** Recent works have shown great success in learning latent representations that are capable of coupling visual signals with language. One such example is CLIP [32] model. CLIP uses a contrastive training scheme [6] on over 400 million pairs of images and their captions to build a shared latent representation for visual-textual content. In this space, similarity between images and text can be computed. It has been shown that CLIP is an effective tool for image generation tasks. For example, providing natural-language guided manipulation of human face imagery in StyleGAN space [30, 26, 1]. Similarly, CLIP can be employed in 3D: using differentiable rendering, images of a 3D scene are rendered in a forward pass and are scored against a text prompt with CLIP. Taking the similarity score as an optimization objective, gradients are back-propagated through the CLIP network and the rendering process, back to the underlying representation of the 3D scene. Text2Mesh [27] uses CLIP to optimize for colors and positions of mesh vertices to match a text query. CLIP-Mesh [20] followed suit and proposed to use CLIP to guide normal and texture maps. This basic recipe has been applied to a variety of scene representations, such as point clouds [28] and Neural Fields [38, 18, 39, 13]. In the context of human faces and bodies, CLIP is used for generation of animation sequences [21, 36, 16, 35, 10, 2].

**Zero-Shot 3DMM Shape Reconstruction.** Recovering accurate, explicit meshes from 2D signals such as images or video is an under-constrained and over-parameterized objective. The low-dimensional underlying representation of 3DMMs proves useful not only for modeling, but for 3D reconstruction of geometry from images of humans. The parameters can serve as a data-driven regularization term for shape and pose, and the low dimensional representation convexifies the optimization problem of fitting such a model to an image, rather than a freely deforming mesh. Recent 3DMM-from-image methods either use iterative optimization schemes to fit the parameters of the 3DMM to the image [23] or directly regress the shape and pose parameters from an Image [12, 8, 22, 19]. Our method allows to use CLIP image and text mapping and simply feed-forward the scores through an MLP network to get a body shape.

## 3. Method

An overview of our method can be seen in Figure 2. In essence, our method defines a non-linear mapping from semantic space to parametric space by training a neural network to predict 3DMM coefficients from a vector of semantic attribute scores. We first create a dataset of rendered images of randomly sampled shapes (3.1), gather their corresponding semantic scores (3.2). Then, we employ a scheme to reduce the semantic descriptors to a subset of least-correlated descriptors (3.3) and then train the network (3.4).

| Coverage / Overlap | | | | | |
|---|---|---|---|---|---|
| Method | Algorithm Choice | 2 descriptors | 5 descriptors | 10 descriptors | 15 descriptors |
| SMPLX-male | (6) / 92.1% / 64.5% | 38.6% / 30.1% | 51.5% / 71.4% | 49.7% / 83.5% | 98.3% / 80.5% |
| SMPLX-female | (4) / 49.7% / 56.7% | 27.1% / 49.5% | 50.0% / 67.1% | 47.4% / 83.0% | 82.4% / 83.4% |
| SMPLX-neutral | (8) / 52.0% / 83.0% | 40.2% / 51.2% | 50.0% / 67.1% | 51.9% / 86.9% | 52.1% / 90.3% |
| SMPL-male | (4) / 97.5% / 60.2% | 40.1% / 46.7% | 97.7% / 66.5% | 98.3% / 83.9% | 98.2% / 89.3% |
| SMPL-female | (5) / 95.7% / 68.6% | 36.3% / 47.3% | 95.7% / 68.6% | 97.8% / 85.2% | 99.1% / 89.9% |
| SMPL-neutral | (6) / 96.1% / 76.0% | 80.4% / 48.7% | 96.2% / 71.5% | 99.6% / 85.2% | 99.8% / 90.8% |
| FLAME-expression | (3) / 13.4% / 56.8% | 10.4% / 47.4% | 19.3% / 67.2% | 27.0% / 77.3% | 34.4% / 85.0% |

Table 1: Choosing different number of descriptors for a model. We evaluate the coverage and overlap of vertices for a mapper that was trained by using different number of descriptors (see Section 5.1, the threshold used to measure coverage is 0.3). The number chosen by our algorithm is shown in parentheses at the beginning of each row. As can be seen, the more descriptors are chosen the larger the cover is, but the larger the overlap between the descriptors is as well.

### 3.1. Dataset Creation

The process of creating the data for training our model is similar across all 3DMMs (we demonstrate it on four different ones in this paper). For each model, we only use the first 10 principal components, that cover above 95% of the variance. We draw $N_{samples}$ 10-dimensional random vectors of coefficients. For body models, we draw a random shape coefficients vector $\vec{\beta}$, which yields a random shape. From the expression basis of the FLAME model, we randomly sample an expression coefficient vector $\vec{\psi}$ which yields a random expression. Since random values may also lead to noisy and unrealistic outputs, we limit the values of the sampled coefficients to a range of $k$ standard deviations of the model's coefficients ($k = 2$ for body shapes and $k = 4$ for face expressions), yielding more realistic results. The other parameters of the 3DMM $(\theta, \gamma)$ remain neutral.

Defining the semantic representation begins by gathering an over-complete set $\ell$ of $N_\ell$ word descriptors that correspond to each 3DMM. Our method supports any set of words that relate to the model: we couple the body model with body descriptors, and choose face descriptors for the face model. For the body models SMPL and SMPL-X, we adopt the set of descriptors used in [34], for FLAME face expression we use an expanded set based on [9], and for FLAME face shape we used our own set of descriptors. Lastly, since the SMAL model was trained on 5 animal families (Felidae-big cats, Canidae-dogs, Equidae-horses, Bovidae-cows, Hippopotamidae-hippos) we used a set of descriptors that contains animals from these given families. The full list of descriptors can be found in the supplemental file. We note that our method can be used with any desired set of relevant descriptors.

### 3.2. CLIP Ratings

We use CLIP [32] to encode each text descriptor $\ell_i$ using their text encoder $e^{\ell_i} = CLIP_{Text}(\ell_i)$. Each mesh created from a random coefficient vector $F(\vec{\beta}, \theta, \gamma)$ is rendered to create a set of images $I_j$, and these images $I_j$ are encoded by CLIP image encoder $e^{I_j} = CLIP_{Image}(I_j)$ to the same latent space of the encoded text (see Figure 2). In this space, we can compute the compatibility between the encoded image $e^{I_j}$ and each encoded label $e^{\ell_i}$ using cosine similarity to get a score $\Omega(I_j, \ell_i) = \cos(e^{I_j}, e^{\ell_i})$. This way, each random coefficients vector $\vec{\beta}$ or $\vec{\psi}$ is paired with a vector of the descriptors' scores:

$$\Omega_{i=1}^{N_\ell} = [\Omega(I_j, \ell_1), \ldots, \Omega(I_j, \ell_{N_\ell})] = [\Omega_1, ..., \Omega_{N_\ell}]$$

containing $N_\ell$ scores.

### 3.3. Descriptors Selection

There are many words that could describe a body shape, face shape or facial expression. As the score of a given image and word descriptor in CLIP's embedding space is calculated by their semantic similarity, there might be many possible word descriptors for a single image, and many possible images for a single word descriptor. For example, both the words "happy" and "smile" would have a high correlation with a smiling face, and therefore, their effect on a facial expression is entangled. Similarly, "tall" and "short" are entangled in a body shape model. Our key observation is that a large number of descriptors are difficult to handle for interactive modeling, and the larger the number, the more entangled they are in terms of their effect on the shape – making it even harder to achieve the desired shape results (see Table 1).

Contrary to BodyTalk [34], our goal is to use a minimal set of semantic word descriptors, while covering as much of the PCA shape space of each 3DMM as possible. We present an algorithm for selecting a subset of semantic descriptors with two main objectives: they should cover a large part of the shape space, and they should be disentangled as much as possible.

Our idea is to first segment the shape space into regions, and assign descriptors to each region. This segmentation assures that we cover the shape space as we do not neglect any region. Next, we select for each region a set of descriptors by checking their variance (i.e., promoting coverage), and removing correlated descriptors, leaving the remaining ones as disentangled as possible.

**Clustering.** To segment the shape space, we use our $N_{samples}$ dataset as a proxy to represent a sampling of the space and cluster their images (created as described in Section 3.1) encoded to CLIP's latent space. we use K-means algorithm with Silhouette score to find $K$.

Next, we compare all encoded images $e^I$ in each cluster $C_i$ to all $N_\ell$ encoded word descriptors $e^{\ell_i}$ for $i = 1, \ldots, N_\ell$. The top 5 most similar descriptors in a cluster are given a vote for this cluster. We normalize the value of votes by the size of each cluster. Then, each word descriptor is assigned exclusively to the cluster with the largest sum of votes for this descriptor.

**Choosing descriptors.** To anticipate the effect of a given descriptor on the 3DMM, we use the variance of its score $\Omega(I', \ell_i)$ on all images $I'$ in the dataset. The larger the variance, the more descriptive this descriptor will be in terms of shape variation. To anticipate the level of entanglement between descriptors, we use their correlation. For each descriptor $\ell_i$ we build a vector of $N_{samples}$ scores $\Omega(I', \ell_i)$ for all images $I'$ in the dataset. Correlation is measured between these vectors. The larger the correlation between two descriptors, the more similar the effect of this descriptor is on the shape, and the more entangled they are (we show some correlation plot examples in the supplemental file). Lastly, we apply antonyms and synonyms detection to verify there are no such pairs in our final set of descriptors.

The process of choosing the descriptors is first performed on each cluster separately, and then merging the lists. We sort the cluster descriptors according to their variance in descending order. Thus, the first descriptor that is chosen from each cluster is the descriptor with the highest variance. Once a descriptor is chosen, we iterate over the other descriptors that are left for this cluster and filter out the ones that have a high correlation with the chosen descriptor (we use the median correlation value of all pairs as the threshold). In addition, before adding the next descriptor to the set, we check if its synonyms or antonyms are already chosen, and if so, we skip it. This process continues until the list of descriptors is exhausted.

Lastly, we merge the lists created from all clusters and sort again according to the descriptor variance. In a similar manner as we did for each cluster, we filter from the merged list all correlated descriptors, synonyms and antonyms. Finally, we arrive at a set of $d$ chosen semantic descriptors for this model. $d$ can be different for each model, but our algorithm can also support any preset number of descriptors by removing descriptors from the bottom of the list or adding back descriptors that were filtered out according to their variance order (see Table 1).

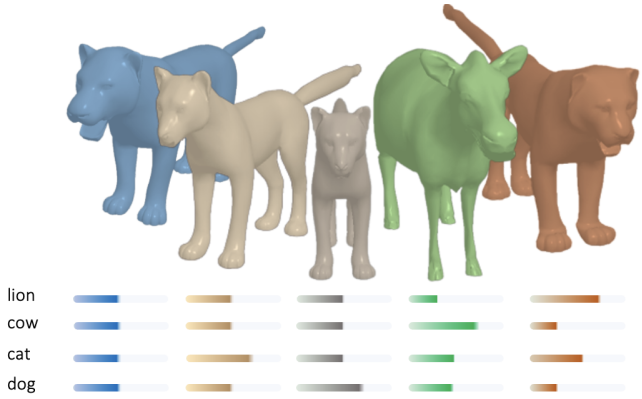


Figure 3: Models created using an interactive slider application for SMAL[45] model (the blue model on the left is the neutral model). These animals were created by interpolating between four semantic descriptors - lion, cow, cat and dog. As can be seen, other animals could also be designed as well as intermixed creatures.

Our algorithm can also support a preset list of descriptors if the user seeks to fit a 3D model with respect to some specific descriptors. In this case, the process described above for choosing the final set of descriptors is simply initialized with the user pre-defined descriptors, so that they are always contained in the final set.

### 3.4. Training

For simplicity, we will denote the coefficients vector by $\vec{\xi}$ for both shape and expression parameters. Our goal is to define the mapping from the semantic representation of the $d$ descriptors to the parametric representation represented by $\vec{\xi}$. From our dataset creation (Section 3.1), each coefficients vector $\vec{\xi} = [\xi_1, ..., \xi_{10}]$ is paired with a scores vector $\vec{\Omega} = [\Omega_1, ..., \Omega_{N_\ell}]$. Hence, for a given scores vector $\vec{\Omega}$, the goal is to predict the corresponding coefficients vector $\vec{\xi}$. This mapping resembles the one presented in [34], only we rely on CLIP score to label our data, providing a single score rather than multiple scores obtained by crowd sourcing, and we do not assume a linear relationship between the word descriptors and the 3DMM's coefficients vector. Instead, we train a multi-layer-perceptron with ReLU activations as the mapping function using our paired data and a simple $L_2$ loss $\mathcal{L} = ||\hat{\vec{\xi}} - \vec{\xi}||_2$. The network consists of hidden layers (500 and 800 neurons), and is trained for 50 epochs.

# 4. Applications

## 4.1. Interactive Sliders

Using the set of semantic descriptors $\ell$ chosen by our algorithm, we can build a simple and intuitive interface to control the 3DMM. This is because the descriptors are least entangled and cover the PCA space well. First, we train a mapper $M$ with the chosen set of descriptors $\ell$ as described above (Section 3.4). Next, we create a slider for each descriptor whose value represents the desired CLIP score for that specific descriptor (see Figure 4). The user can now change each slider to trigger a re-prediction of the 3DMM coefficients. The computation is interactive as the interface takes all sliders' values as input $\vec{\Omega}$ to $M$, which maps them in a forward pass to the 3DMM coefficients vector $\vec{\xi}$ and renders the corresponding 3D mesh. Thus, the modeler can interact and understand the relationships between the descriptors and their effect on the mesh. Figure 3 demonstrates examples of 3D meshes of different animals that were created using an interface created for the SMAL model.

## 4.2. Zero-Shot Image to Shape Reconstruction

Our method allows to leverage CLIP's semantic understanding to define a zero-shot image to shape reconstruction. Given any image of a person, we embed it into CLIP's latent space and use the similarity scores of the image against the set of descriptors $\ell^M$ chosen for a mapper $M$ to obtain the vector $\vec{\Omega}$. Next, we simply feed $\vec{\Omega}$ through the network to get the 3DMM coefficients vector $\vec{\xi}$ that best fits the image and create the shape from $\vec{\xi}$. Our method differs from state-of-the-art methods in that it enables the user to refine the zero-shot prediction very simply using semantic sliders starting from a relatively good initial guess.

# 5. Experiments

## 5.1. Coverage Evaluation

To evaluate a mapper $M$ that was trained with a given set of descriptors $\ell$, we would like to measure its effect on the 3DMM. For each descriptor $\ell_i \in \ell$ we do a forward pass through the mapper twice: the first pass by setting a low score to $\Omega_i$ and the second pass by setting a high score to $\Omega_i$. The other coefficients are set to a default value and remain constant. Each such pass creates a parameter vector $\vec{\xi} = [\xi_1, ..., \xi_{10}]$, which represents the resulting 3DMM shape or expression. We denote these vectors as $\vec{\xi}(i)_{low}$ and $\vec{\xi}(i)_{high}$.

We measure the geometric effect of a descriptor $\ell_i$ on a 3DMM by examining the deformation of the vertices of the mesh. This is evaluated by comparing the position of each vertex in the two extreme cases of $\vec{\xi}(i)_{low}$ and $\vec{\xi}(i)_{high}$. Hence, for each vertex $v$ the size of deformation is defined as $\delta'(v) = ||v_{low} - v_{high}||_2$. We can normalize these values by $\delta_{\max} = \max_{v \in Mesh} \delta'(v)$ and get a value $\delta(v) = \delta'(v)/\delta_{\max}$ between 0 and 1 (see Figure 5).

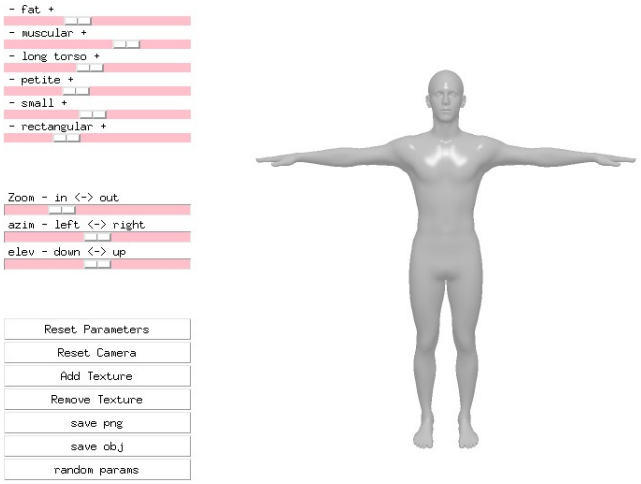


Figure 4: A simple and intuitive interactive application can be defined using sliders for each semantic descriptor. With Semantiy, the sliders have strong semantic meaning and are disentangled so that changing one will have very limited affect on others.

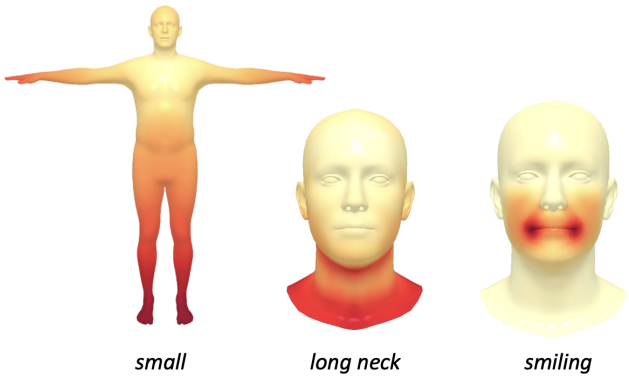


Figure 5: The geometric effect of a given descriptor. The darker the color of the vertex is (more red), the more it is affected by the descriptor. To define coverage we collect all vertices with a value larger than a given threshold $\theta$.

Using these differences, we can evaluate the coverage of a specific descriptor by measuring the number and positions of vertices it affects in the mesh. For example, we can look at all vertices $v$ whose value is above a threshold $\delta(v) > \theta$ for any given descriptor. We can also compare the overlap between two descriptors by measuring the intersection-over-union of their covered vertices. We use these measure

in our ablation studies as described next in Section 5.2. Note that this measure is only an approximation as it measures the size of the change of vertices and not their direction.

### 5.2. Ablations

To further evaluate our algorithm's performance, we conducted experiments that include training the mapper with different numbers of descriptors and training on different sizes of training data. As we do not have valid ground truth to estimate the accuracy of our mapper we created a test set that contains several body shapes that were generated by the first 10 blendshapes $\vec{\xi}$ of the SMPL-X model. In each experiment we optimize the descriptors' values $\vec{\Omega}$, to minimize the error $||\hat{\vec{\xi}} - \vec{\xi}||_2$. The goal of these experiments is to test the expressive power of the given configuration to match a given shape. The smaller the error the more expressive the method is. We show the results of these ablations in the supplemental file. We used this evaluation to choose the hyper-parameters we use in our algorithm: training with 3K sample images with texture for all models.

### 5.3. Interactive Sliders User Study

To evaluate Semantify's performance we conducted a user study that compares our approach against BodyTalk [34], as well as the original basis of the 3DMM. The user is presented with a target 3D model (randomly created using the 3DMM) and is tasked to fit a 3DMM mesh model to the target shape using a set of sliders, using the two compared interfaces (see Figure 4) at random order. Users were given up to 5 minutes to complete the task. BodyTalk was implemented on the SMPL body model, therefore we compare it to our SMPL male and female models. In another study, we compare our SMPLX male and female models to the original PCA-based axes. In both studies, we seek to evaluate the accuracy of the user-fitted 3DMMs, the time that it took to fit the model, as well as the overall subjective experience of users. Since we cannot obtain the outputs of BodyTalk in the first study, we asked humans to rate the results, and determine which of them fits more accurately to the input shape. We had 10 users perform both studies (5 males, 5 females aged between 23 to 54), 7 of which are novices and 3 of which were professional 3D modelers. We present the quantitative results of these studies in Table 2. As can be seen, Semantify achieves better accuracy in less time. Note that theoretically only the baseline can reach zero error, but because it is very difficult to handle, the error obtained was higher than Semantify. In addition, we asked the users for feedback on the experience of using the different applications (A was ours, B was the alternative). Here are some examples of such reviews (more examples could be found in the supplemental file):

"Using application B, occasionally when changing one slider it affects the other, which causes the user to start all over again".

"Using application B every minor change in a certain slider generated major changes in the rest of the sliders"

"Application A was far friendlier and I sensed as though I maintained much more control over the different body features"

"The abundance of sliders on application B only made it harder to control, not the other way around".

### 5.4. Zero Shot Reconstruction

To measure our zero-shot 3D-shape reconstruction method of SMPLX model, we use HBW (Human Bodies in-the-wild) dataset presented in SHAPY [7], containing ground-truth 3D body scans with in-the-wild images. Table 3 shows the performance of our method compared to SHAPY and PIXIE [12] on all of HBW validation set. Note that SHAPY also takes keypoints (usually obtained from openPose [5]) as input. Some examples are shown in Figure 6. As can be seen, our zero-shot performance are on-par with state of the art methods, but we have the advantage of using semantic descriptors. This allows the user to fine tune the results from a very good initial guess, simply by using the semantic sliders. Some examples results with around a minute of fine-tuning can be found in the supplemental file.

## 6. Conclusion

We have presented a method to edit a 3D parametric model using semantically meaningful descriptors by first choosing a subset of descriptors judiciously and then learning a non-linear mapping from this set to the 3DMM coefficients. All this is done in a self-supervised manner harnessing the abilities of CLIP visual-language model. Our method can support a more intuitive user interface that allows even novices to model 3D shapes. It also supports other applications such as zero-shot reconstruction and fine-tuning of shapes from images.

**Limitations and future directions.** There are several limitations to our method. First, our mapper's performance relies heavily on the data that is created (Section 3.1). For example, a dataset must contain extreme samples to achieve a more expressive mapper. Second, our goal with Semantify was to build a semantic representation of a given 3DMM without a human-in-the-loop. However, in some cases manually tuned modeling for a specific 3DMM would probably improve the 3DMM mapper's performance (we further elaborate on this in the supplemental file). Lastly, as shown in many other works, CLIP has rich semantic understanding, but there are cases in which its performance degrades. For example, performance is better with textured meshes, so using various textures for each 3DMM may improve the

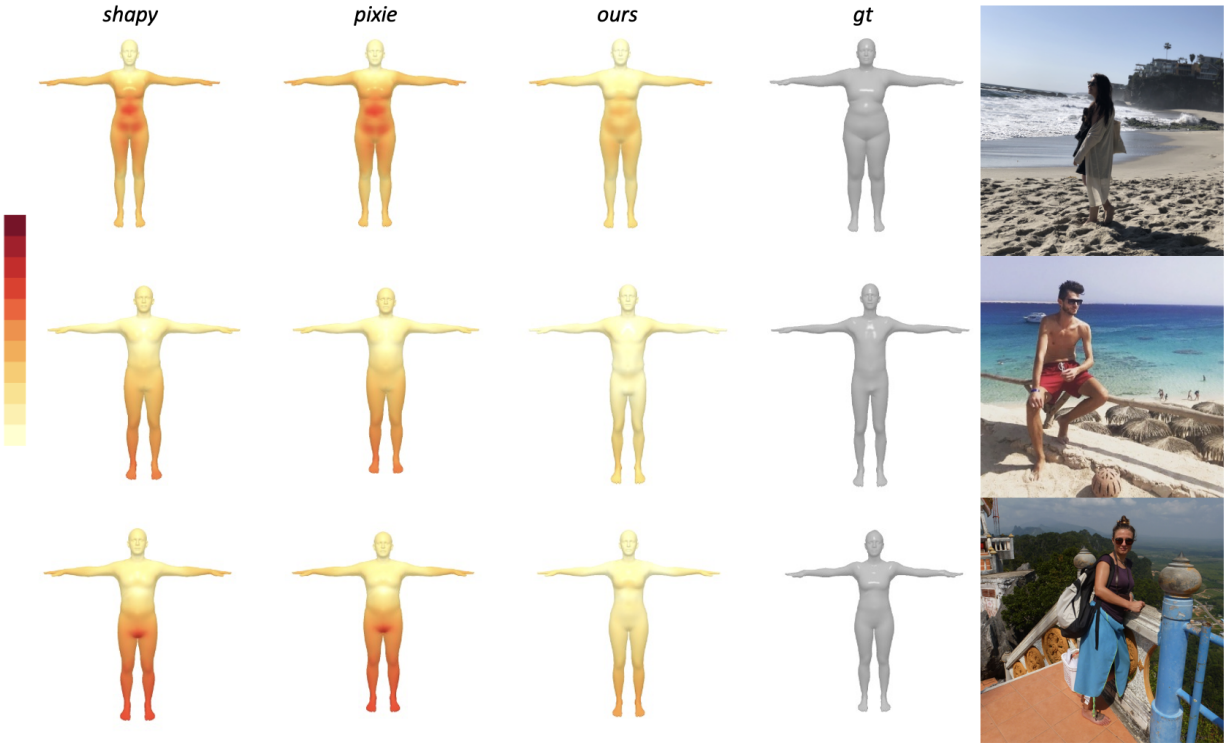


Figure 6: Results of zero-shot image to shape reconstruction. We compare Semantify against SHAPY and PIXIE on SMPLX 3DMM. Colors indicate error compared to ground truth (darker means larger error).

| | Time (min) | | | |
|---|---|---|---|---|
| | Males | Females | Total | Score |
| BodyTalk | 3.22 | 3.16 | 3.19 | 0.04% |
| ours | **2.12** | **1.45** | **1.59** | **0.96%** |

| | Mean Error (cm) | | | Time (min) | | |
|---|---|---|---|---|---|---|
| | Males | Females | Total | Males | Females | Total |
| Baseline | 0.122 | 0.111 | 0.116 | 3.18 | **2.09** | 2.43 |
| ours | **0.121** | **0.099** | **0.110** | **2.22** | 2.17 | **2.19** |

Table 2: User-study results for the SMPL 3DMM compared against BodyTalk [34] (left, Score means the percent that this model was selected as better fitting the input), and SMPL-X parametric model compared against the raw blendshapes (right).

| | Males | Females | Total |
|---|---|---|---|
| SHAPY | **0.0196934** | **0.014422** | **0.017085** |
| PIXIE | 0.024779 | 0.015265 | 0.020324 |
| Semantify (ours) | 0.022632 | 0.016564 | 0.019666 |

Table 3: Comparing our zero-shot shape reconstruction from image method against SHAPY[7] and PIXIE [12]. The values demonstrate the average MSE over HBW's in-the-wild validation set.

mapper's performance. Furthermore, more subjective descriptors such as "cute", "scary", "pretty" etc., are harder to rate using CLIP, and would probably not work well in our model.

Future work can also improve the zero-shot 3D shape reconstruction by using prior information about its context (for example, the gender of a person, body keypoints as used in [7] etc.). Adding such priors along with CLIP's semantic understanding may surpass the existing state-of-the-art solutions.

## 7. Acknowledgments

This research was supported by the Israel Science Foundation (grant No. 1390/19). We would also like to thank Eyal Gomel for valuable discussions.

## References


[1] Rameen Abdal, Peihao Zhu, John Femiani, Niloy J. Mitra, and Peter Wonka. Clip2stylegan: Unsupervised extraction of stylegan edit directions. *CoRR*, abs/2112.05219, 2021.

[2] Shivangi Aneja, Justus Thies, Angela Dai, and Matthias Nießner. Clipface: Text-guided editing of textured 3d morphable models. *arXiv preprint arXiv:2212.01406*, 2022.

[3] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. Scape: Shape completion and animation of people. *ACM Transactions on Graphics (TOG)*, 24:408–416, 07 2005.

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

[5] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43:172–186, 2018.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations, 2020.

[7] Vasileios Choutas, Lea Müller, Chun-Hao P. Huang, Siyu Tang, Dimitrios Tzionas, and Michael J. Black. Accurate 3d body shape regression using metric and semantic attributes. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[8] Vasileios Choutas, Georgios Pavlakos, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. Monocular expressive body regression through body-driven attention. In *European Conference on Computer Vision (ECCV)*, 2020.

[9] Radek Daněček, Michael J. Black, and Timo Bolkart. Emotion driven monocular face capture and animation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20311–20322, June 2022.

[10] Ginger Delmas, Philippe Weinzaepfel, Thomas Lucas, Francesc Moreno-Noguer, and Grégory Rogez. Posescript: 3d human poses from natural language. In *European Conference on Computer Vision*, pages 346–362. Springer, 2022.

[11] Bernhard Egger, William AP Smith, Ayush Tewari, Stefanie Wuhrer, Michael Zollhoefer, Thabo Beeler, Florian Bernard, Timo Bolkart, Adam Kortylewski, Sami Romdhani, et al. 3d morphable face models—past, present, and future. *ACM Transactions on Graphics (TOG)*, 39(5):1–38, 2020.

[12] Yao Feng, Vasileios Choutas, Timo Bolkart, Dimitrios Tzionas, and Michael Black. Collaborative regression of expressive bodies using moderation. In *International Conference on 3D Vision (3DV)*, pages 792–804, Dec. 2021.

[13] Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8649–8658, June 2021.

[14] Thomas Gerig, Andreas Morel-Forster, Clemens Blumer, Bernhard Egger, Marcel Luthi, Sandro Schoenborn, and Thomas Vetter. Morphable face models - an open framework. *2018 13th IEEE International Conference on Automatic Face amp; Gesture Recognition (FG 2018)*, May 2018.

[15] Matthew Hill, Stephan Streuber, Carina Hahn, Michael Black, and Alice O'Toole. Exploring the relationship between body shapes and descriptions by linking similarity spaces. *Journal of Vision*, 15(12):931–931, 2015.

[16] Fangzhou Hong, Mingyuan Zhang, Liang Pan, Zhongang Cai, Lei Yang, and Ziwei Liu. Avatarclip: Zero-shot text-driven generation and animation of 3d avatars. *arXiv preprint arXiv:2205.08535*, 2022.

[17] IEEE. *A 3D Face Model for Pose and Illumination Invariant Face Recognition*, Genova, Italy, 2009.

[18] Ajay Jain, Ben Mildenhall, Jonathan T. Barron, P. Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 857–866, 2021.

[19] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

[20] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. *SIGGRAPH Asia 2022 Conference Papers*, 2022.

[21] Youwang Kim, Ji-Yeon Kim, and Tae-Hyun Oh. Clip-actor: Text-driven recommendation and stylization for animating human meshes. In *European Conference on Computer Vision*, 2022.

[22] Muhammed Kocabas, Chun-Hao P. Huang, Joachim Tesch, Lea Müller, Otmar Hilliges, and Michael J.

Black. SPEC: Seeing people in the wild with an estimated camera. In *Proc. International Conference on Computer Vision (ICCV)*, pages 11035–11045, Oct. 2021.

[23] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *ICCV*, 2019.

[24] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017.

[25] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Smpl: A skinned multi-person linear model. *ACM Trans. Graph.*, 34(6), nov 2015.

[26] Xudong Lou, Yiguang Liu, and Xuwei Li. Tecm-clip: Text-based controllable multi-attribute face image manipulation. In *Proceedings of the Asian Conference on Computer Vision*, pages 1942–1958, 2022.

[27] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13492–13502, 2022.

[28] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *ArXiv*, abs/2212.08751, 2022.

[29] Ahmed A. A. Osman, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. Supr: A sparse unified part-based human representation. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 568–585, Cham, 2022. Springer Nature Switzerland.

[30] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021.

[31] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019.

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

[33] Hyewon Seo, Frederic Cordier, and Nadia Thalmann. Synthesizing animatable body models with parameterized shape modifications. 07 2003.

[34] Stephan Streuber, M. Alejandra Quiros-Ramirez, Matthew Q. Hill, Carina A. Hahn, Silvia Zuffi, Alice O'Toole, and Michael J. Black. Body Talk: Crowdshaping realistic 3D avatars with words. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 35(4), July 2016.

[35] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. *arXiv preprint arXiv:2203.08063*, 2022.

[36] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Daniel Cohen-Or, and Amit H Bermano. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[37] Ayush Tewari, Michael Zollhöfer, Pablo Garrido, Florian Bernard, Hyeongwoo Kim, Patrick Pérez, and Christian Theobalt. Self-supervised multi-level face model learning for monocular reconstruction at over 250 hz. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[38] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2022.

[39] Can Wang, Ruixia Jiang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Nerf-art: Text-driven neural radiance fields stylization. *ArXiv*, abs/2212.08070, 2022.

[40] Jinbao Wang, Shujie Tan, Xiantong Zhen, Shuo Xu, Feng Zheng, Zhenyu He, and Ling Shao. Deep 3d human pose estimation: A review. *Computer Vision and Image Understanding*, 210:103225, 2021.

[41] Lizhen Wang, Zhiyua Chen, Tao Yu, Chenguang Ma, Liang Li, and Yebin Liu. Faceverse: a fine-grained and detail-controllable 3d face morphable model from a hybrid dataset. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2022)*, June 2022.

[42] Hongyi Xu, Eduard Gabriel Bazavan, Andrei Zanfir, William T Freeman, Rahul Sukthankar, and Cristian Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6184–6193, 2020.

[43] Haotian Yang, Hao Zhu, Yanru Wang, Mingkai Huang, Qiu Shen, Ruigang Yang, and Xun Cao. Facescape: a large-scale high quality 3d face dataset and detailed riggable 3d face prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[44] Silvia Zuffi, Angjoo Kanazawa, and Michael J. Black. Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3955–3963, 2018.

[45] Silvia Zuffi, Angjoo Kanazawa, David Jacobs, and Michael J. Black. 3D menagerie: Modeling the 3D shape and pose of animals. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, July 2017.