Received: 8 February 2021 | Revised: 10 May 2021 | Accepted: 11 May 2021

DOI: 10.1002/cav.2028

RESEARCH ARTICLE

WILEY

Local control editing paradigms for part-based 3D face morphable models

Donya Ghafourzadeh[1] | **Sahel Fallahdoust**[1] | **Cyrus Rahgoshay**[1] | **Andre Beauchamp**[1] | **Adeline Aubame**[1] | **Tiberiu Popa**[2] | **Eric Paquette**[3]

[1]La Forge, Ubisoft, Montreal, Quebec, Canada

[2]Computer Science and Software Engineering, Concordia University, Montreal, Quebec, Canada

[3]Software Engineering and IT, École de technologie supérieure, Montreal, Quebec, Canada

Correspondence
Donya Ghafourzadeh, Ubisoft La Forge, Montreal, QC, Canada.
Email: gh.dony@gmail.com

Abstract

We propose an approach to construct realistic 3D facial morphable models (3DMM) that allows an intuitive facial attribute editing workflow. Current face modeling methods using 3DMM suffer from a lack of local control. We thus create a 3DMM by combining local part-based 3DMM for the eyes, nose, mouth, ears, and facial mask regions. Our local principal component analysis (PCA)-based approach uses a novel method to select the best eigenvectors from the local 3DMM to ensure that the combined 3DMM is expressive, while allowing accurate reconstruction. We provide different editing paradigms, all designed from the analysis of the data set. Some use anthropometric measurements from the literature and others allow the user to control the dominant modes of variation extracted from the data set. Our part-based 3DMM is compact, yet accurate, and compared to other 3DMM methods, it provides a new trade-off between local and global control. We tested our approach on a data set of 135 scans used to derive the 3DMM, plus 19 scans that served for validation. The results show that our part-based 3DMM approach has excellent generative properties and allows the user intuitive local control.

KEYWORDS

computer graphics, computing methodologies, mesh models, shape modeling

1 | INTRODUCTION

The authoring of realistic 3D faces with intuitive controls is used in a broad range of computer graphics applications, such as video games, person identification, facial plastic surgery, and virtual reality. This process is particularly time-consuming, given the intricate details found in the eyes, nose, mouth, and ears. Consequently, it would be convenient to use high-level controls, such as anthropometric measurements, to edit human-like character heads.

Many methods use 3D morphable face models (3DMM) for animation (blend shapes), face capture, and face editing. Even though face animation concerns are important, our work focuses on the editing of facial meshes. 3DMMs are typically constructed by computing a principal component analysis (PCA) on a data set of scans sharing the same mesh topology. New 3D faces are generated by changing the relative weights of the individual eigenvectors. These methods are popular due to the simplicity and efficiency of the approach, but suffer from two fundamental limitations: they impose global control on the new generated meshes, making it impossible to edit a localized region of the face, and the control mechanism is very unintuitive. Some methods compute localized 3DMMs but those focus on facial animation instead of

face modeling. We compared our approach to previous works relying on facial animation and saw that their automatic localized basis construction works well for animation purposes (considering a data set composed of animations for a single person), but performs worse than our approach for modeling purposes (considering a data set made of neutral faces from different persons).

We propose an approach to construct realistic 3DMMs. We increase the controllability of our faces by segmenting them into independent sub-regions and selecting the most dominant eigenvectors per part. Furthermore, we rely on facial anthropometric measurements to derive useful controls to use in our 3DMM for editing faces. We propose a measurement selection technique to bind the essential measurements to the 3DMM eigenvectors. Our approach allows the user to edit faces by adjusting the facial parts using sliders controlling the values of anthropometric measurements. The measurements are mapped to eigenvector weights, allowing us to compute the individual parts matching the values selected by the user. The reconstructed parts are seamlessly blended together to generate the desired 3D face. We also provide local editing freedom beyond anthropometric measurements and ensure to generate realistic faces while satisfying user constraints. Finally, we verify the realism of 3D facial models and demonstrate how realistic they are. Our main contributions can be summarized as:

- A process to select how to distribute the budget of eigenvectors across the multiple parts;
- A process to select anthropometric measurements;
- An online editing approach allowing interaction through three strategies to deform the face (measurement sliders, handles, vertex-based);
- A validation system (face proportions, angles, nose depth).

2 | RELATED WORK

3D morphable models are powerful statistical models widely used in many applications in computer vision and computer graphics. One of the most well-known previous works in this regard is that by Blanz and Vetter.[1] Their pioneer work proposes a model using PCA from face scans. Although they propose a multi-segment model and decompose a face into four parts to augment expressiveness, the PCA decomposition is computed globally on the whole face. Other global PCA methods have been proposed.[2-7] A downside of global PCA-based methods is that they exhibit global support: when we adjust the eye, the nose may also undergo undesirable changes. Another downside is a lack of intuitive user control for face editing. While the eigenvectors are good at extracting the dominant modes of variation of the data, they provide weak intuitive interpretation.

To address the former problem, local models have been proposed. They segment the face into independent sub-regions and select the most dominant eigenvectors per part. Tena et al.[8] propose a method to create localized clustered PCA models for animation. They select the location of the basis using spectral clustering on the geodesic distance and a correlation of vertex displacement considering variations in the expressions. Their method requires a manual step to adjust the boundaries of the segments, making it somewhat similar to ours, where the parts are user-specified. Chi et al.[9] adaptively segment the face model into soft regions based on user-interaction and coherency coefficients. Afterwards, they estimate the blending weights which satisfy the user constraints, as well as the spatio-temporal properties of the face set. Here too, the required user intervention renders the segmentation somewhat similar to our user-provided segments. SPLOCS[10] propose the theory of sparse matrix decompositions to produce localized deformation from an animated mesh sequence. They use vertex displacements in the Euclidean coordinates to select the basis in a greedy fashion. We noticed that when considering variation in identity instead of variation in expression, the greedy selection leads to bases which are far less local than those obtained from both our method and Tena et al.'s.[8] These papers address facial animation instead of face modeling and therefore assume large, yet localized deformations caused by facial expressions, which are different from our context where each face is globally significantly different from the others.

Like Tena et al.,[8] Cao et al.[11] segment the face with the same spectral clustering, followed by manual adjustment. While their method focuses mostly on expression, they also provide some identity modeling, as they rely on the FaceWarehouse[3] global model, which they decompose using the segments defined by spectral clustering. In their case, the goal is to adapt a 3DMM to a face from a video feed, in real time. While their method works remarkably well for the real-time "virtual makeup" application, it lags behind ours in terms of providing a very detailed facial model, and it does not support a face editing workflow.

Other papers supplement decomposition approaches with the extraction of fine details, allowing to reconstruct a faithful facial model.[12-14] The major problem with these approaches is that they work for a specific person and do not provide editing capabilities. The Phace[15] method allows the user to edit fat or muscle maps in texture spaces on the face. While this provides a physically based adjustment, the control is implicit. The user modifies the texture and then the system simulates muscles and fat to get the result.

Wu et al.[16] propose an anatomically constrained local deformation model to improve the fidelity of monocular facial animation. Their model uses 1000 overlapping parts, and then decouples the rigid pose of the part from its nonrigid deformation. While this approach works particularly well for reconstruction, the parts are too small for editing semantic face parts such as the nose or the eyes.

Contrary to the methods described thus far, the Allen et al.[2] and BodyTalk[17] methods greatly facilitate editing by mapping intuitive features to modifications of global 3DMM eigenvector weights. In particular, BodyTalk[17] relates transformations of the meshes to keywords such as "fit" and "sturdy." While the mapping between the words and the deformations is not perfect, it still makes it reasonably intuitive to edit the mesh of the body. One problem with this method is that it provides words for bodies, not faces. A second major problem is the inability to make local adjustments, and adjustments that increase the length of the legs will result in changes to other regions such as the torso and arms. In contrast, for our approach, we aim at providing local control in the editing.

A downside of global PCA-based methods is that they exhibit global support: adjusting parameters to change one part has unwanted effects on other unrelated parts. To address this problem, our approach segments the face into independent sub-regions and provides a process to select the best set of eigenvectors, given a target number of eigenvectors. Other methods that segment the face in sub-regions target facial animation instead of modeling. We demonstrate that our approach is better-suited to the task of face editing than these methods. Another problem with most of the previous related works is that they do not allow facial model editing through the adjustment of objective measurements. In contrast, our approach relies on anthropometric measurements used as controls for editing. Furthermore, we propose a process to select the right set of anthropometric measurements for each facial part. This article extends our previous work[18] in the area of local editing which is complementary to anthropometric measurements. We provide more degrees of freedom for the user and ensure to generate realistic facial features besides satisfying the deformation prescribed by the user. We also propose a validation system that measures the realism of faces and verifies the compliance to many measurements from the literature.

3 | OVERVIEW

In this article, we introduce a pipeline for constructing a 3DMM. We separate the face into regions and compute independent PCA decomposition on each region. We then combine the per-region 3DMMs, paying particular attention to the selection of the most dominant eigenvectors across the eigenvectors of the different regions. While the eigenvectors are good at extracting the dominant data variation modes, they provide weak intuitive interpretation. We thus use anthropometric measurements to provide human understandable adjustments of the face. The reconstruction from the measurements is done through a mapping from the measurements to the weights that need to be applied to each eigenvector. From the set of measurements, we extracted from our survey of the literature, we selected a subset which resulted in the least reconstruction error. An overview of our approach can be found in Figure 1. The remainder of this article is organized as follows: Section 4 describes how 3DMMs are constructed, including face decomposition and selection of the most dominant eigenvectors. Afterwards, we discuss how to reconstruct a face by smooth blending of different facial parts (Section 5). In Section 6, the selection of the anthropometric measurements, and the mapping between these measurements and the PCA eigenvectors are discussed. We propose an interactive geometry local editing in Section 7 to complement anthropometric measurements. We demonstrate the results in Section 8, and discuss them in Section 9.

4 | 3D MORPHABLE FACE MODEL

We employ PCA on a data set of faces to construct our 3DMMs. All faces are assumed to share a common mesh topology, with vertices in semantic correspondence. We propose to segment the face into different parts in order to focus the decomposition on a part-by-part basis instead of computing the PCA decomposition on the whole face. We compute the decomposition separately for the male and female subsets. As shown in Figure 1, we decompose the face into five parts:

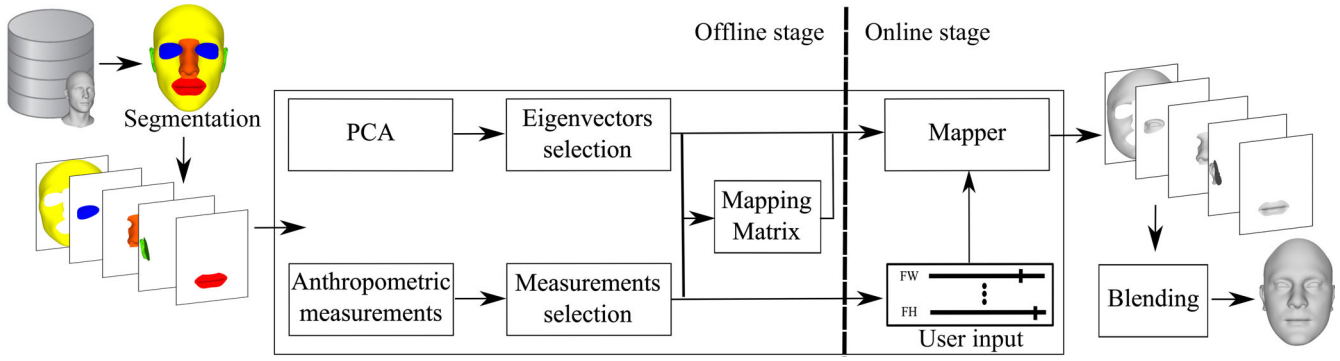


FIGURE 1 Our 3D facial morphable model workflow. In an offline stage, we extract PCA eigenvectors and select the best ones. We also select the best subset of anthropometric measurements. The relationship between the eigenvectors and measurements is encoded in a mapping matrix. All of these are used in the online stage, where the mapper collects user-prescribed anthropometric measurement values, and applies the mapping matrices to reconstruct the parts. The last step provides the edited face through a smooth blending of the parts

eyes, nose, mouth, ears, and what we refer to as the facial mask (which groups the remaining areas such as cheeks, jaws, forehead, and chin). We further discuss this design choice in Section 9.3. This face decomposition allows us to have eigenvectors for each part. The geometry of the facial parts is represented with a shape-vector $S_d = \left[V_1 \ \dots \ V_{n_v}\right] \in R^{3n_v}$, where n_v is the number of vertices of dth facial part, $d \in \{1, \dots, 5\}$, and $V_i = [x_i y_i z_i] \in R^3$ defines the x, y, and z coordinates of the ith vertex. After applying PCA, each facial part d is reconstructed as

$$S'_d = \overline{S_d} + \sum_{j=1}^{n_e} P_j b_j, \tag{1}$$

where $\overline{S_d}$ is the mean shape of dth facial part, n_e is its number of eigenvectors, P_j is an eigenvector of size $3n_v$, b is a $n_e \times 1$ vector containing the weights of the corresponding eigenvectors, and S'_d is the reconstruction, which will be an approximation when not using all eigenvectors.

Our approach selects the smallest set of eigenvectors that still reconstructs the shape accurately. We accomplish this by incrementally adding the eigenvectors, in the order of their significance, to the reconstruction until a certain accuracy is met. Even though we rely on the eigenvalues to sort the eigenvectors for each part (largest to smallest eigenvalue), we provide the user with a measurable error (in mm), which is more precise than relying solely on eigenvalues across different parts. We determine the best set of eigenvectors to achieve a balance between the quality of the per-part reconstruction and the whole face reconstruction. To evaluate the accuracy of our selection, we construct the facial parts (Equation 1) and blend them together (Section 5) to generate the whole face. Afterwards, we assess the accuracy of the reconstruction by calculating the average of the geometric error D_{GE} between the ground truth and the blended face. In order to avoid rigid motions in our evaluation, we first do a rigid alignment step (rotation and translation) between the facial parts of the ground truth and the blended result. We calculate the rotation based on the vertices of the boundary and then calculate the translation of the rotated part considering all of its vertices. In the case of the eyes and ears, we consider them independently for the left side and right side. We then record the average per-vertex Euclidean distance over all vertices and per part:

$$D_{GE_{all}}(S') = \frac{1}{n_{all}} \sum_{d=1}^{5} \sum_{\substack{V'_j \in S'_d \\ V_j \in S_d}} ||V_j - (R_d V'_j + T_d)||, \tag{2}$$

$$D_{GE_{part}}(S') = \frac{1}{5} \sum_{d=1}^{5} \frac{1}{n_d} \sum_{\substack{V'_j \in S'_d \\ V_j \in S_d}} ||V_j - (R_d V'_j + T_d)||, \tag{3}$$

$$D_{GE}(S') = \frac{D_{GE_{all}}(S') + D_{GE_{part}}(S')}{2}, \tag{4}$$

TABLE 1 Number of eigenvectors selected for each part

Facial part	# Eigenvectors for female	# Eigenvectors for male
Facial mask	7	9
Eye	10	5
Nose	6	6
Mouth	9	10
Ear	14	16

where n_{all} is the number of vertices of the face mesh, n_d is the number of vertices for part d, V_j is on the ground truth, and V'_j is the corresponding point on the blended face. We compute averages over all vertices and per part to ensure that parts with more vertices do not end up using most of the eigenvector budget at the expense of parts with fewer vertices. We do so for the entire data set and for a set of 19 validation faces that were not part of the training data set. We compute the median data set error as well as the median validation error, and we average the two in a global error. The process of reconstructing the parts of validation faces is done by projecting each part onto the corresponding eigenvector basis (followed by the blending process).

At each step of our incremental eigenvector selection, we decide which of the five parts will get a new eigenvector added to its set. We compare the geometric errors resulting from each of the five candidate eigenvectors, and we select a candidate eigenvector which has a great impact on decreasing the error. When eigenvectors from multiple parts result in decreases of error, instead of systematically picking the eigenvector based on lowest error, we select by sampling from a discrete probability density function (PDF) created from the respective decreases in error of the five candidate eigenvectors. Each part with a decrease of error is assigned a probability corresponding to the value of the decrease for this part divided by the sum of the decreases for all parts. Parts with an increase of error are assigned a probability of zero and do not count toward the sum. This PDF selection process creates a more even distribution of eigenvectors across the parts and maintains a low error.

As we iterate, the reconstruction error decreases. For the female and male data set faces, the average reconstruction errors are 2.00 and 2.13 mm when considering zero eigenvectors. The errors decrease to 0.75 and 0.74 mm after 80 iterations, and when considering all eigenvectors the errors are 0 mm. We chose an error threshold of 1 mm which balances out the cost associated with considering too many eigenvectors and the accuracy of the reconstruction. Table 1 shows the resulting eigenvector distribution after achieving our 1 mm reconstruction accuracy. We experimented with reconstructing the female and male validation faces based on using our subset of eigenvectors. The median reconstruction errors are 1.33 and 1.48 mm, respectively.

5 | FACE CONSTRUCTION THROUGH PARTS BLENDING

This step focuses on the problem of constructing a realistic new face by blending the five segmented parts together. As opposed to methods such as those of Tena et al.[8] and Cao et al.,[11] which handle the transition between the parts by relying on the adjustment of a single strip of vertices, we spread the transition across three strips of vertices. In contrast to other methods that adjust the transition by vertex averaging[11] or least-squares fitting,[8] we use Laplacian blending[19] of the parts and the transition, resulting in a smooth, yet faithful global surface. The vertex positions are solved by an energy minimization which reduces the surface curvature discontinuities at the junction between the parts while maintaining the desired surface curvature. To this end, we define a transition zone made of quadrilateral strips around the parts. In our experiments, a band of two quadrilaterals (three rings of green vertices in Figure 2) provides good results. We interpolate the Laplacian $\mathscr{L}$ (the cotangent weights) of the five facial parts weighted by β_d, which has values of $\beta_d = 1$ inside the part, $\beta_d \in \{0.75, 0.5, 0.25\}$ going outward of the part in the transition zone, and $\beta_d = 0$ elsewhere. We normalize these weights such that they sum to one for each vertex. These soft constraints allow some leeway in the transition zone. The boundary conditions of our system are set to the ring of blue vertices in Figure 2, and we solve for the remaining vertices. To this end, we minimize the following energy function:

$$E(V') = \sum_{i \in \text{inner}} \left\| T_i \mathscr{L}(V'_i) - \frac{\sum_{d=1}^{5} \beta_{i,d} R_d \mathscr{L}(V_{i,d})}{\sum_{b=1}^{5} \beta_{i,b}} \right\|^2, \tag{5}$$

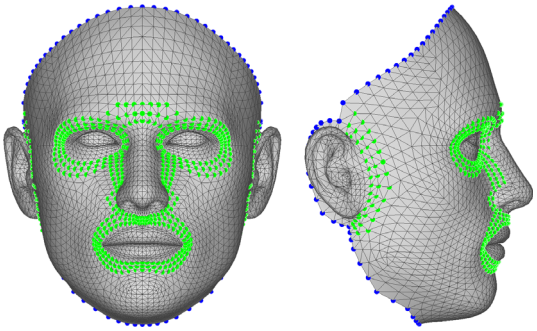


FIGURE 2 Regions highlighted in green and blue contain the transition and fixed zones, respectively

where "inner" is the set of vertices of the five parts, excluding the vertices of the boundary conditions; T_i is an appropriate transformation for vertex V_i' based on the eventual new configuration of vertices V_i and R_d is the rotation of part d.

We solve Equation (5) in a similar fashion to ARAP[20] by alternating solving for the vertex position and rotation matrices until the change is small. Figure 3 shows that the rotation quickly converges as the Frobenius norm of consecutive rotations is large only for the first few iterations. Given our experiments, we decided to stop iterating when the Frobenius norm fell below 0.01 or after six iterations. Figure 4 shows an example of a blended face. In this case, the Frobenius norm was below 0.01 after five iterations. Figure 5 shows the evolution of the geometric error D_{GE} between the ground truth parts and their blended counterparts for the example of Figure 4. As can be seen, the error quickly reaches a plateau as the rotation stabilizes.

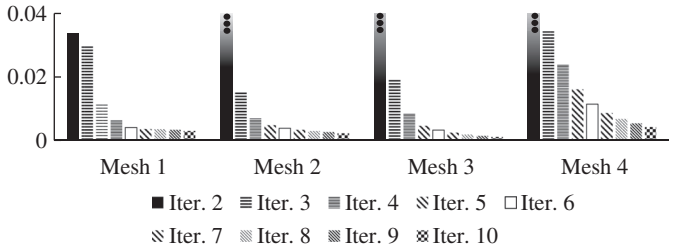


FIGURE 3 Graph showing the evolution of the Frobenius norm of the rotation between two consecutive iterations (averaged across the five rotations R_d). For each of the meshes 1–4, we begin with the average parts and change the weight of one eigenvector per part. Each eigenvector is selected randomly (from the first 10 eigenvectors if there are more than 10 eigenvectors for the part). The new value for the weight is also randomly selected within the range of −2 and +2 times the standard deviation for this eigenvector

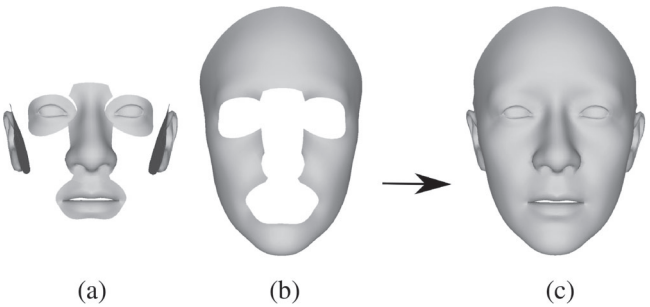


FIGURE 4 (a, b) The generated parts. As in Figure 3, we modified each average part by changing the weight of one eigenvector selected randomly. The new value for the weight is also randomly selected. (c) The result of blending the facial parts

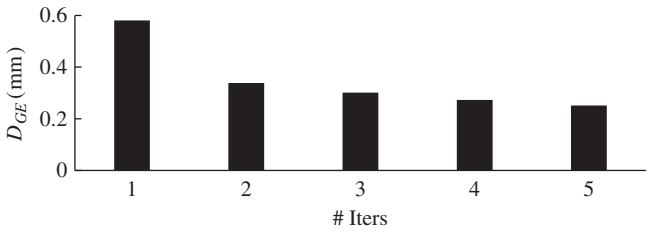


FIGURE 5 Graph comparing the average geometric error (in mm) between the ground truth parts and their blended counterparts, for different numbers of iterations

6 | SYNTHESIZING FACES FROM ANTHROPOMETRIC MEASUREMENTS

PCA eigenvectors characterize the data variation space, but do not provide a clear intuitive interpretation. In this article, we focus mainly on constructing linear regression models from data using a set of intuitive facial anthropometric measurements. Facial anthropometric measurements provide a quantitative description by means of measurements taken between specific surface landmarks defined with respect to anatomical features. We use the 33 parameters listed in Table 2. Each measurement corresponds to either a Euclidean distance or a ratio of Euclidean distances between surface positions, as specified in each paper cited in Table 2. In this section, we propose a measurement selection technique which assesses the accuracy of each measurement, resulting in the most relevant ones for each facial part.

6.1 | Mapping method

We compute the measurements on the facial parts of the training data set, yielding $f_{d,i} = \left[f_{i_1} \ \dots \ f_{i_{n_m}}\right]$ for the dth facial part of scan S_i considering n_m measures. The measures for all of the scans are combined into an $n_m \times n_s$ matrix, $F_d = \left[f_{d,1}^T \ \dots \ f_{d,n_s}^T\right]$, where n_s is the number of scans. We want to determine how to adjust the weights of the PCA eigenvectors to reconstruct faces having specific characteristics corresponding to the measures. We adopt the general method of Allen et al..[2] However, while that method computes a global mapping that adjusts the whole body, we will compute per-part local mappings. Furthermore, in Section 6.2, we will derive a process to select the best measures out of the set of all measures $\left[f_{i_1} \ \dots \ f_{i_{n_m}}\right]$ and will proceed independently for each of the five parts.

We relate measures by computing a linear mapping to the PCA weights. With the n_m measures for the dth facial part, the mapping will be represented as a $(n_e) \times (n_m + 1)$ matrix, M_d:

$$M_d[f_{i_1} \ \dots \ f_{i_{n_m}} 1]^T = b, \tag{6}$$

where b is the corresponding eigenvector weight vector. Collecting the measurements for the training data set, the mapping matrix is solved as:

$$M_d = B_d F_d^+, \tag{7}$$

where B_d is a $(n_e) \times (n_s + 1)$ matrix containing the corresponding eigenvector weights of the related facial part and F_d^+ is the pseudoinverse of F_d. As in Equation (6), a row of 1s is appended to the measurement matrix F_d for y-intercepts in the regression.

To construct a new facial part based on specific measurements, we use b in Equation (1), as follows:

$$S'_d = \overline{S_d} + Pb, \tag{8}$$

where $\overline{S_d}$ is the mean shape of the dth facial part and P is the matrix containing the eigenvectors. Moreover, we can define delta-feature vectors of the form:

$$\Delta f_d = [\Delta f_1 \ \dots \ \Delta f_{n_m} 0]^T, \tag{9}$$

where each Δf contains the user-prescribed differences in measurement values. Afterwards, by adding $\Delta b = M_d \Delta f_d$ to the related eigenvector weights, it is possible to adjust the measure such as to make a face slimmer or fatter.

6.2 | Measurement selection

We propose a novel technique for automatically detecting the most effective and relevant anthropometric measurements. Some might be redundant with respect to others, some might not make sense for a specific part (e.g., the "Ear Height" might not be relevant for the mouth), and some might even lead to mapping matrices that generate

TABLE 2 Anatomical terms and corresponding abbreviations of our selected and discarded measurements

Anatomical term	Abbrev	Ref
Selected measurements		
Nasal Width ÷ Root Width	NWRW	21
Nasal Width ÷ Length of Bridge	NWLB	21
Nasal Width ÷ Width of Nostril	NWWN	21
Nasal Root Width ÷ Tip Protrusion	NRTP	21
Length of Nasal Bridge ÷ Tip Protrusion	NBTP	21
Nasal Width ÷ Tip Protrusion	NWTP	21
Nasal Root Width ÷ Length of Bridge	NRLB	21
Nasal Root Width ÷ Width of Nostril	NRWN	21
Length of Nasal Bridge ÷ Width of Nostril	NBWN	21
Width of Nose ÷ Tip Protrusion	WNTP	21
Philtrum Width	PW	22
Face Height	FH	23
Orbits Intercanthal Width	OIW	23
Orbits Fissure Length	OFL	23
Orbits Biocular Width	OBW	23
Nose Height	NH	23
Face Width	FW	24
Bitragion Width	BW	24
Ear Height	EH	24
Bigonial Breadth	B	25
Bizygomatic Breadth	BB	25
Facial Index	F	26
Nasal Index	N	26
Mouth-Face Width Index	MFW	26
Biocular Width-Total Face Height Index	BWFH	26
Lip Length	LL	27
Maximum Frontal Breadth	Max FB	27
Interpupillary Distance	ID	27
Nose Protrusion	NP	27
Nose Length	NL	27
Nose Breadth	NB	27
Discarded measurements		
Eye Fissure Index	EF	26
Minimum Frontal Breadth	Min FB	27

worst results. During our investigations, we discovered that considering more anthropometric measurements does not necessarily lead to a lower reconstruction average error. Figure 6 illustrates that a higher error occurs considering all the measurements in comparison with our selected combination. In order to aggregate the error for the given part, we reconstruct the face, relying only on the anthropometric measurements of the selected part, and then calculate the average error, as in Section 4. Figure 7 shows two odd-looking examples from using all the measurements of the nose (Figure 7(a)) and facial mask (Figure 7(b)). We thus evaluate the set of relevant measurements, separately for each part. We begin with an empty set of *selected measurements*, and we iteratively test which

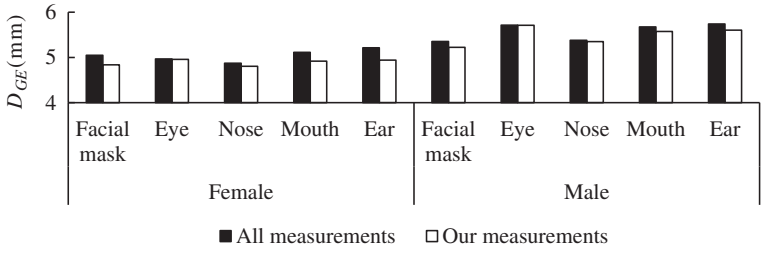


FIGURE 6 Using all measurements leads to higher reconstruction errors (mm) as compared to our set of selected measurements on the data set and validation faces

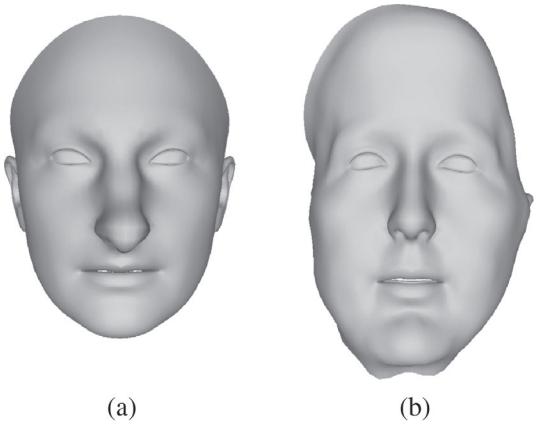


FIGURE 7 Using all the semantic measurements of the nose (a) and facial mask (b) often leads to odd-looking parts when editing through the adjustment of measurement values

measurement we should add to the set by evaluating the quality of the reconstructed faces when creating the mapping matrix, considering the currently selected measurements together with the *candidate measurement*. We reconstruct a face using the mapping matrix (Equations 6 and 8) based only on its measurement values. The reconstructed face is considered as a *prediction*, and thus we evaluate the prediction quality in a fashion very similar to that used for eigenvector selection, by reconstructing all of the faces found in the data set of facial scans, as well as the 19 validation faces.

Each candidate measurement is used together with the current set of selected measurements, and we compute the candidate mapping matrix from this set of measurements (using the training data set). We use the mapping matrix on the training and validation data set, and reconstruct all of the instances of the part under consideration (e.g., all of the mouths). We then evaluate a geometric error, D_{GE} (Equation 4), with the per-vertex distance between each predicted instance and its corresponding ground truth instance. Note that we evaluate the whole face by reconstructing only the desired part (e.g., the mouth) and use the parts of the ground truth for the rest of the face. The distance is calculated after a rigid alignment of the predicted instance to the ground truth instance is performed. We can thus ensure that we are evaluating the fidelity of the shape, and not its pose. If one or a few faces result in a large error, this could lead to the rejection of a measurement, which might still be beneficial for the prediction of most faces. To avoid this, we also measure the percentage D_{NI} of faces for which an error improvement is seen. We count the number of faces whose geometric errors have been decreased by considering the candidate measurement. We then normalize D_{GE} and D_{NI} to the $[0, 1]$ range and combine them into a single reconstruction quality measure:

$$quality = normalize(D_{NI}) + 1 - normalize(D_{GE}). \tag{10}$$

Considering the combined geometric error and percentage of improvement of all candidate measurements, we pick the one which will be added to the set of selected measurements. We stop adding measurements when we observe all of candidates measurements lead to an increase of D_{GE} and a value D_{NI} below 50%. We repeat this process for each part (eyes, nose, mouth, etc.)

The selected anthropometric measurements are enumerated in Table 3. The description of each measurement, as well as the reference to the literature from which we obtained the measurement, are shown in Table 2, where we also list the measurements we rejected (measurements which were never selected for any of the segments).

TABLE 3 Combinations of anthropometric measurements

Part	Selected measures
For female	
Facial mask	B, BB, BW, BWFH, FH, MaxFB, NBWN, NH, NP, PW, WNTP
Eye	B, BB, BW, BWFH, EH, F, FW, ID, LL, NB, NBTP, NH, NRLB, NRTP, NWRW, NWWN, OBW, OFL, OIW, PW
Nose	EH, LL, N, NB, NBTP, NBWN, NH, NL, NP, NWTP, NWWN, PW
Mouth	BWFH, F, LL, MFW, NP, NRWN, NWTP, PW
Ear	EH, FH, FW, MaxFB, NB, NBTP, NBWN, NP, NRLB, NRTP, NWLB, NWTP, OBW, PW, WNTP
For male	
Facial mask	BW, BWFH, F, FH, FW, MaxFB, NRWN, OBW, OFL, PW
Eye	B, BB, BW, F, FW, ID, N, NBTP, NBWN, OBW, OFL, PW
Nose	BB, FW, ID, LL, MaxFB, NB, NBWN, NH, NP, NRLB, NRTP, NWLB, NWWN, OBW, OIW
Mouth	B, BW, BWFH, F, LL, NBTP, NH, PW
Ear	B, BB, BWFH, EH, FH, MaxFB, N, NRTP, NWRW, OBW, OIW

6.3 | Correlation between measurements

Defining the correlation between the measurements is important for the adjustment of faces. Accordingly, if the user adjusts one measurement, the system automatically calculates the adjustment of the other measurements as well. This greatly helps to create realistic faces by maintaining the correlation observed in the data set. Similarly to Body Talk,[17] we use Pearson's correlation coefficient on F to evaluate the relationship between the anthropometric measurements. Considering a facial part d, the Pearson's correlation coefficient Cor_{jk} for measurements j and k is expressed as:

$$\mathrm{Cor}_{jk} = \frac{\sum_{i=1}^{n_s}(f_{ij} - \overline{f_j})(f_{ik} - \overline{f_k})}{\sqrt{\sum_{i=1}^{n_s}(f_{ij} - \overline{f_j})^2}\sqrt{\sum_{i=1}^{n_s}(f_{ik} - \overline{f_k})^2}}, \tag{11}$$

where $f_{ij}, f_{ik} \in f_{d,i}$ are measurements of scan S_i, $\overline{f_j}$ and $\overline{f_k}$ are the mean values of measurements j and k, respectively, and n_s is the number of scans. The coefficient is a value between -1 and 1 that represents the correlation. When adjusting measurement k by Δf_k, we get the change in other measures as $\Delta f_j = \mathrm{Cor}_{jk}\Delta f_k$. Accordingly, we can evaluate the influence of one measurement on the others, as well as the conditioning on one or more measurements, and create the most likely ratings of the other measurements.

7 | EDITING PARADIGMS FROM DEFORMATION MODES

Complementary to the anthropometric measurements, we propose two editing approaches where the user anchors the editing at specific locations on the mesh. The handle-based editing provides the user with a visualization of the locations on the mesh corresponding to the vertices that move the most with respect to eigenvectors. The vertex-based editing allows the user to select the location where the deformation will take place and our approach uses the eigenvectors to derive a deformation mostly located in the selected region and matching the displacement desired by the user. Being based on the eigenvectors, these two approaches make sure to generate realistic facial features beside satisfying the deformation prescribed by the user.

7.1 | Handle-based editing

We propose handles as a visual cue for editing facial features based on high-variance eigenvectors. We use the first 10 eigenvectors and we find the vertex with the largest magnitude for each of them. Each of these vertices is a candidate

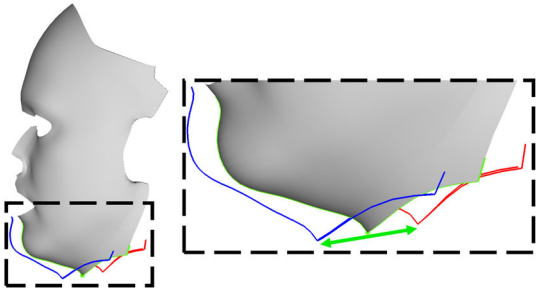


FIGURE 8 Visualization of the deformation related to an eigenvector. Blue and red silhouettes show the deformation related to weights corresponding to −2 and +2 standard deviation

handle. The main idea is to select handles that lead to a big deformation. Consequently, we disregard the handles when the magnitude of the corresponding vertex movement is smaller than half of the average edge length of the 1-ring neighborhood. We let the user visualize the handles with a line anchored at the vertex. The green arrow in Figure 8 shows the motion of the handle vertex for the first eigenvector of the facial mask. The extent of the motion corresponds to −2 and +2 standard deviations from the mean. The green, red, and blue silhouettes show the mean, −2 and +2 standard deviation, respectively.

To ease the visualization, handles which are close to each other (20 mm) will be displayed from the same vertex. For this purpose, we find the closest vertex to the average of the vertices positions which have a close distance, within 20 mm. We display the direction of each handle using a vector corresponding to the eigenvector deformation anchored at the corresponding vertex. The size of the vector is adjusted between 5 and 10 mm depending on the norm of the related displacement. As faces and eigenvectors are mostly symmetric, for each handle we display a corresponding symmetric handle. Note that both handles have the same effect. When the user interacts with a handle, the amount of pushing or pulling on the handle determines the adjustment of the weight for this eigenvector.

7.2 | Vertex-based editing

As opposed to selecting predetermined handles, the user can also select any vertex and prescribe a translation for this vertex. In an energy minimization manner, we define a Gaussian falloff around that vertex, and vertices with a low Gaussian weight want to remain at their current location, while vertices with a high Gaussian weight are more free to move. As such, the deformation is more localized and we have a smooth transition zone.

The user can move the selected vertex u in any direction, and our approach will aim to find the weights which will meet the prescribed vertex movement and while keeping fixed the regions with low falloff weight. Our approach finds the weight b by solving following energy:

$$E(b) = \lambda|(Pb)_u - x|^2 + |G(Pb - V_0)|^2 + |b - b_0|^2 + |\delta(b - \bar{b})|^2. \tag{12}$$

The first term aims to move vertex u to the position x prescribed by the user (P is the eigenvector matrix, $(Pb)_u$ retrieves the user selected vertex out of the whole mesh Pb, and λ can be adjusted to conform more or less to the user constraint). The second term restricts movement of vertices with low Gaussian falloff (G is a diagonal matrix that weights each vertex according to the Gaussian falloff, V_0 is the vector of previous vertex positions). The third and fourth terms avoid diverging too much from the current b_0 and average $\bar{b}$ weights, respectively. The diagonal matrix δ allows a finer control for each weight. When a weight b is lower than −2 standard deviation or higher than +2 standard deviation, we increase the corresponding entry in δ. This automatically prevents having faces too far from the average. Finally, faces being symmetric, we automatically select the symmetric vertex and apply the corresponding constraints for the mirror vertex movement in Equation (12).

8 | RESULTS

Compared to global 3DMM methods that compute one set of eigenvectors for the whole face, our 3DMM computes a set of eigenvectors for each part. This is at the root of one of the advantages of our approach: its ability to locally adjust faces. We compare our approach to other methods that rely on local 3DMM. We created mapping matrices (Equation 7) for global

3DMMs, SPLOCS,[10] clustered PCA,[8] as well as our part-based 3DMMs, and tested the adjustment of measurements with these models. We used 46 eigenvectors for global 3DMM, SPLOCS, and our part-based 3DMM. For clustered PCA, we first tested using 13 clusters, as is reported in the paper, but found that this leads to a non-symmetrical result (Figure 9(b)). By checking other clusterings, we selected 12 clusters (Figure 9(a)). Because a clustered PCA does not allow for a different number of eigenvectors for each cluster, and to avoid having too few eigenvectors per part, we used 46 eigenvectors for each cluster (selecting the 46 with the largest eigenvalues).

To compare our approach and the use of measurements with other methods, we decided on a way to use our measurements with SPLOCS and clustered PCA. We further demonstrate that with SPLOCS and our approach, we can have more *local measurement* or *global measurement* control. For our approach, Table 3shows that some measures influence more than one part. For example, the "Lip Length" is found in the lists for both mouth and nose. When a measurement is shared between different facial parts, our method allows to decide to have more localized changes by adjusting the measure for only one part, or to have more coherence across the parts by adjusting all of the parts involved in the measurement. If comparing with SPLOCS, we can also balance between local measurements and global measurements. Each measurement is based on computations involving specific *measurement vertices* (such as the corner of the mouth and the tip of the nose). To enforce locality, when considering a measurement, we check which SPLOCS "eigenvectors" infer significant movement at the related *measurement vertices*. We compute this by checking if the eigenvector displacement vector at a *measurement vertex* is large enough as compared to the maximum displacement vector of the eigenvector (we check if it is larger than 1% of the maximum displacement of all vertices of the eigenvector). A SPLOCS eigenvector is considered for a measurement only if it meets the criterion for one of the *measurement vertices* of a specific measurement. To enforce more globality with SPLOCS, we use the mapping matrices for all of the eigenvectors. Figure 10 shows an example of the globality and locality of the influence of adjusting the "Lip Length." It compares global PCA eigenvectors, local measurement and global measurement SPLOCS, clustered PCA, and our local measurement and global measurement approaches. The color coding shows the per-vertex Euclidean distance. Note that the colors do not represent errors, but rather, vertex movements. Thus, the goal is to have warmer colors around the location where

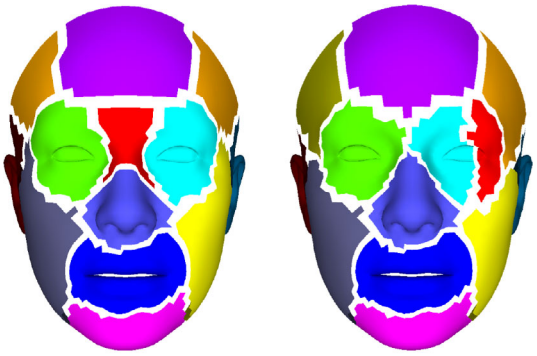


FIGURE 9 Automatic part identification of clustered PCA.[8] Note how the automatic clustering leads to non-symmetrical clusters (left eye with one cluster vs. right eye with two clusters) for 13 clusters and required us to manually check which other clustering would be usable

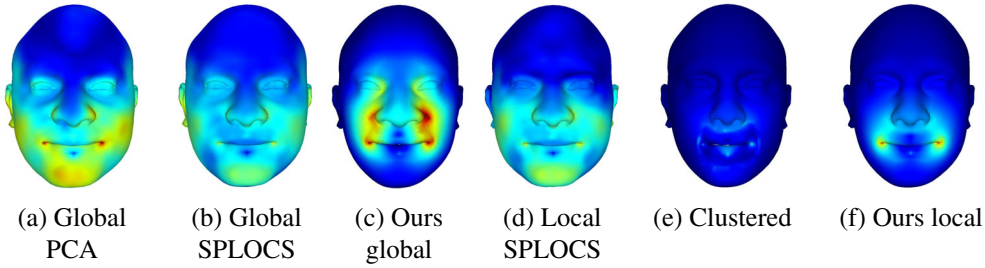


FIGURE 10 Comparison of the globality versus locality of the adjustments (editing by increasing the "Lip Length"): (a) global PCA eigenvectors, (b) global measurement SPLOCS, (c) our global measurement approach, (d) local measurement SPLOCS, (e) clustered PCA, and (f) our local measurement approach. The colors, respectively, represent per-vertex Euclidean distance (blue = 0 mm, red = 8.5 mm). Note how our local measurement and global measurement approaches induce significant and local surface deformation to achieve the desired editing. In comparison, global PCA and SPLOCS induce nonlocal deformation, and clustered PCA induces much less deformation

the editing is intended, and colder colors in unrelated regions. Our method allows having global measurement influenced by adjusting the measure for both the nose and the mouth parts, as well as more localized changes by adjusting only the mouth (Figure 10(c–f)). Contrary to our approach, both global measurement and local measurement SPLOCS resulted in similar deformations all over the face, while the expected result was a modification focused around the mouth (Figure 10(b–d)).

We will now focus on local measurement editing. Figures 11–13 show the adjustment of the same anthropometric measurement using global 3DMM, local measurement SPLOCS, clustered PCA, and our local measurement approach. In Figure 11(f,g), we can see that even though we wanted to adjust the "Nose Breadth," the adjustment using the global eigenvectors and local measurement SPLOCS resulted in significant deformations all over the face, while clustered PCA and our approach could focus the deformation around the nose, as expected (Figure 11(h,i)). We can observe similar unwanted global deformations of the face in Figure 12(f,g). Also note that the automatic segmentation of clustered PCA does not provide the desired deformation for some cases, such as in Figures 12(h) and 13(h). This lack of deformation for clustered PCA (Figures 12(h) and 13(h)) is in part due to the landmarks related to the measurements being close to the boundary of segments. The largest deformation should happen close to the boundary according to the landmark and measurement, but the blending can later diminish the deformation. This effect is worst for clustered PCA because of the larger number of segments (there is always a segment boundary close to a landmark). Accordingly, we consistently outperform clustered PCA in terms of local deformation where expected. In Figure 13, global PCA works reasonably well as the measurement has an influence on a large portion of the face. On the other hand, SPLOCS can deform the expected region, but it is not localized. To summarize, the results shown in Figure 11–13 highlight the difficulty of locally controlling the face deformation, and the power of our approach in locally adjusting the face with respect to the anthropometric measurements.

Figure 14 and the accompanying video show an implementation of our system for measurement-based editing. In the accompanying video, we also show multiple edits on multiple parts, starting from the average face, while Figure 15 shows edits starting from four real faces. We can see that our approach allows capturing the essence of the anthropometric measurements, providing an easy-to-use workflow.

Figures 16 and 17 highlight the fact that our approach is not limited to the measurements and provides more freedom to the user for adjusting facial features. Figure 16 shows how to edit facial features using handles. Figure 16(a) shows the location of the handles. The direction for the handle selected by the user is illustrated in Figure 16(b). The arrows of Figure 16(c) show if the user pulls or pushes the handles. The deformed features are illustrated in Figure 16(d). Figure 17 shows the examples of vertex-based editing where the arrow is showing the vertex selected by the user. Our approach provides a smooth transition across the boundary of the edited regions.

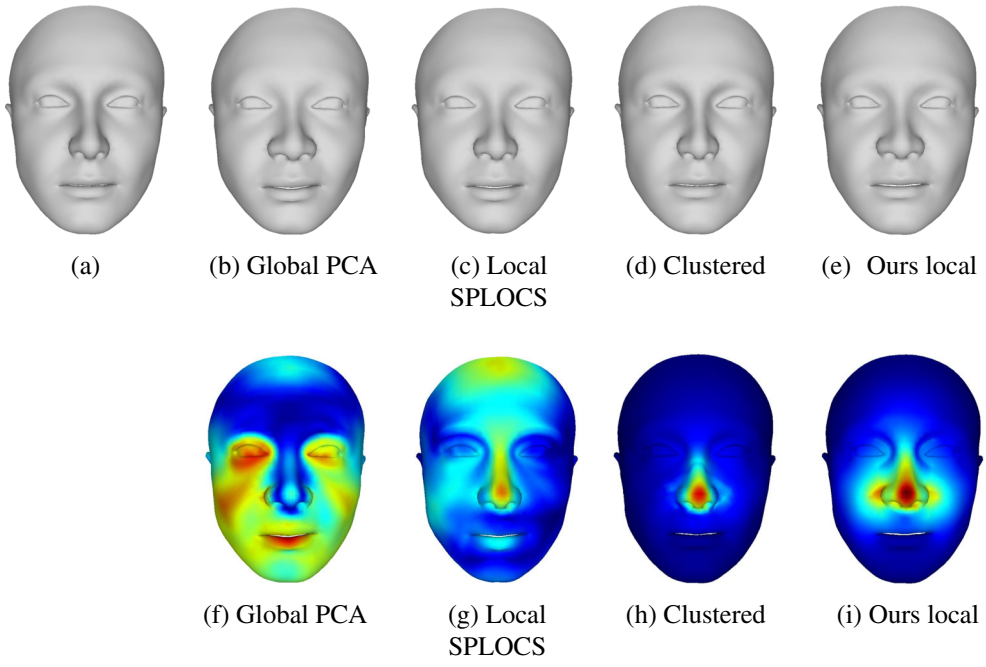


FIGURE 11 "Nose Breadth" adjustment results: (a) nose of a female from validation faces adjusted using (b) global PCA eigenvectors, (c) SPLOCS, (d) clustered PCA, and (e) our approach. The color mapped renderings (f–i) indicate respective per-vertex Euclidean distance (blue = 0 mm, red = 5 mm)

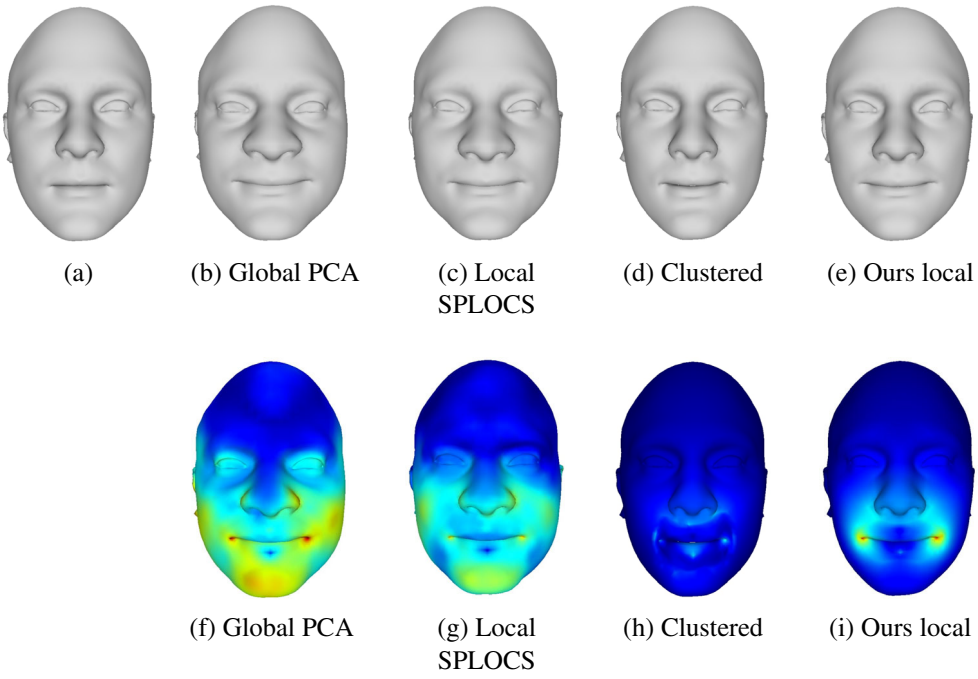


FIGURE 12 "Lip Length" increase results: (a) mouth of a male from validation faces edited using (b) global PCA eigenvectors, (c) SPLOCS, (d) clustered PCA, and (e) our approach. The color mapped renderings (f–i) indicate respective per-vertex Euclidean distance (blue = 0 mm, red = 8 mm)

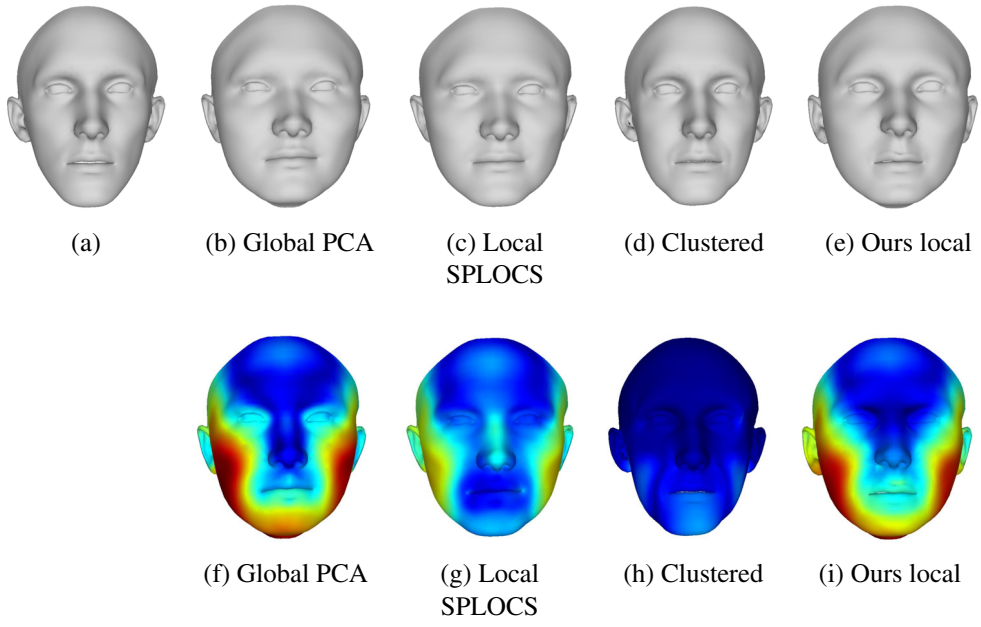


FIGURE 13 "Bizygomatic Breadth" (the bizygomatic width of the face) increase results: (a) A male from the validation faces edited using (b) global PCA eigenvectors, (c) SPLOCS, (d) clustered PCA, and (e) our approach. The color mapped renderings (f–i) indicate respective per-vertex Euclidean distance (blue = 0 mm, red = 14 mm)

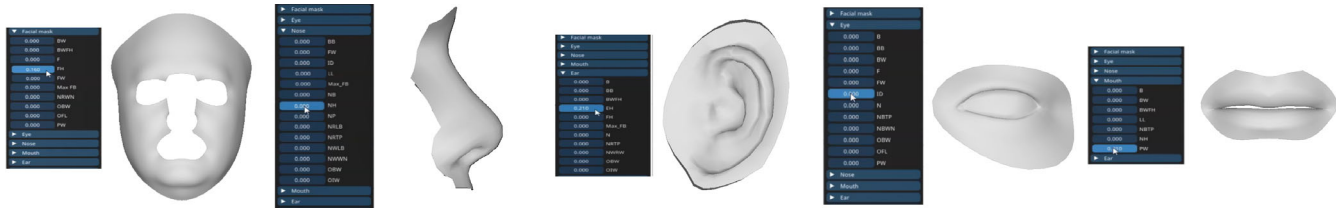


FIGURE 14 Measurement editing system

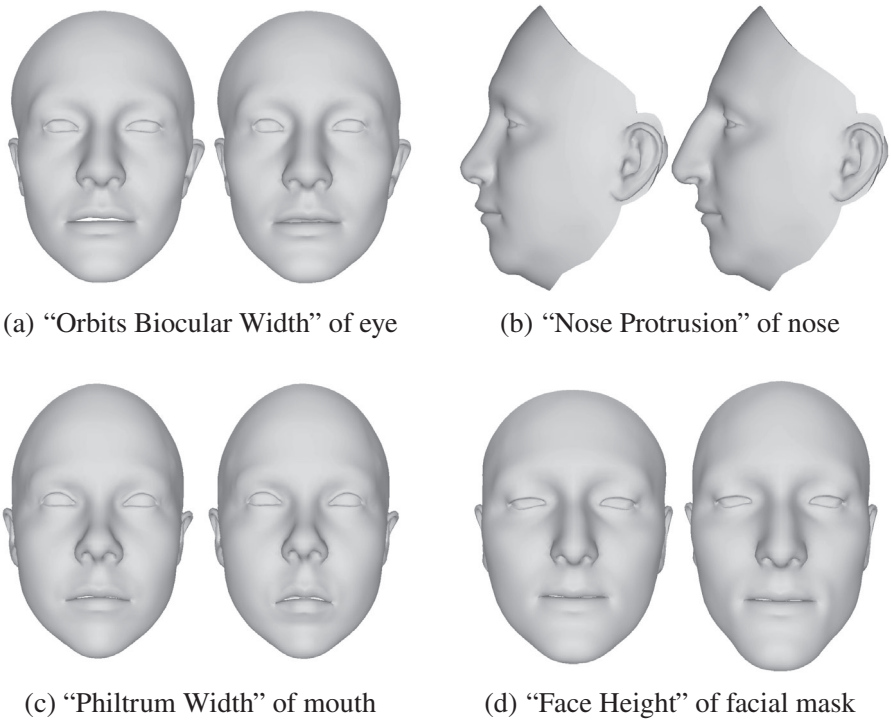


FIGURE 15 We generated random faces (left faces, (a–d)) and edited them by increasing ("+") or decreasing ("–") the value of some of the indicated anthropometric measurements

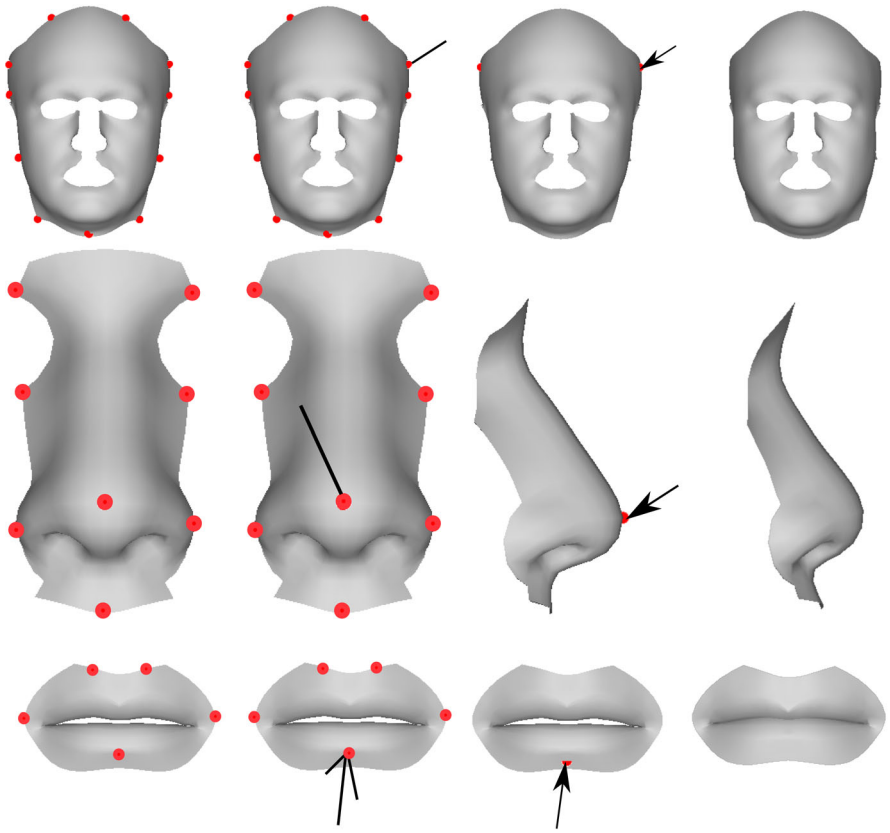
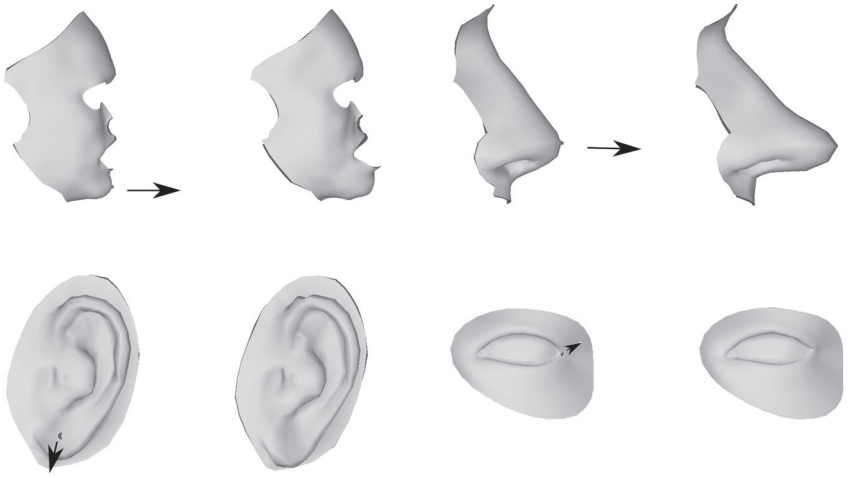


FIGURE 16 Handle-based editing. The arrows show the directions of pulling or pushing of desired handles



(a) Before editing (b) After editing (c) Before editing (d) After editing

FIGURE 17 Vertex-based editing. The arrows show the desired locations

9 | DISCUSSION

In this section, we discuss different aspects of our approach. We present different comparisons highlighting the impact of the eigenvector and measurement selection. We then discuss the face segmentation choice, and end by describing the procedure used to bring all of our scans to a common face mesh.

9.1 | Measurements error

To verify the robustness of our 3DMMs and of our set of selected measurements, we reconstruct real faces, relying on their anthropometric measurements to compute their eigenvector weights (Equation 6). We then get the face with our approach, including the blending procedure (Section 5), and compute its resulting anthropometric measurements. In comparison with Figure 6 which evaluates reconstruction average error, here we compute the quality of the reconstruction through the absolute value of the difference between the ground truth measurement and the measurement from the reconstructed face. Since measurements correspond either to a Euclidean distance or to a ratio of Euclidean distances, we normalized all the measurements to the [0%, 100%] range. Figure 18 shows that the average percentage of error is low when using "our measurements."

This means that both the selection of eigenvectors and the mapping matrix work well. Furthermore, it shows that when using "all measurements" to compute the mapping matrix (Equation 7), we get larger average errors as compared to ground truth measurements. When calculating the error in Figure 18 for "our measurements," we calculate the average error over our selected measurements only (Table 3). The error shown in Figure 18 for "all measurements" also considers only our selected measurements (if the error across all of the measurements is considered, the comparison is even more in favor of using our selected measurements).

We evaluated how our approach compared to SPLOCS and clustered PCA with respect to achieving measurement values prescribed by edit operations. We created a set of 1000 random edits on 135 face meshes. We took the resulting edited face mesh from our approach, SPLOCS, as well as clustered PCA, and evaluate the difference between the measurement value prescribed by the editing and the measurement value calculated from the edited mesh. Overall, our approach is the

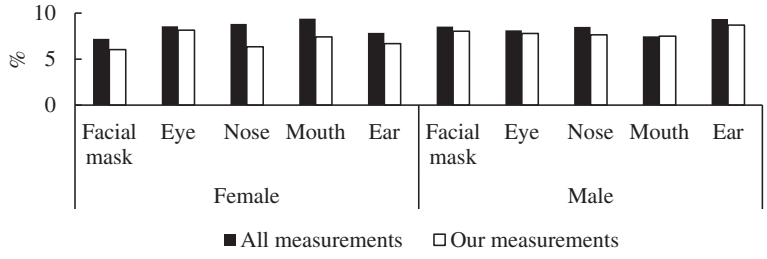


FIGURE 18 Using our subset of measurements on the data set and validation faces leads to lower errors (percentage), as compared to using "all measurements"

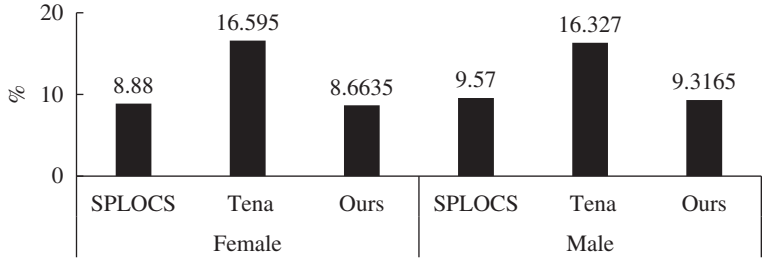
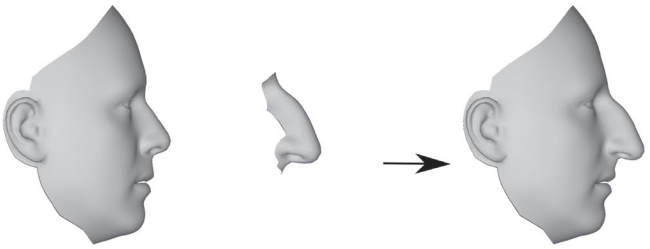


FIGURE 19 Starting from one face, we adjust one of its measurements to match the value of that measurement for another face. We then compute the difference between the prescribed measurement value and the measurement value calculated from the mesh. We do so for 1000 such edits. Our approach leads to a smaller error (percentage), as compared to the clustered PCA, and to slightly better results when compared to local measurement SPLOCS

FIGURE 20 (a) Average male head. Its "Nose Height" is 45.09 mm. (b) A synthesized nose with its "Nose Height" edited to 70.11 mm. (c) Result of blending the nose. While this is an extreme case, it still reflects the fact that the approach is not always able to achieve the prescribed measurement (the value decreased to 58.53 mm for this example)



one that performed the best, with the resulting measurement being closest to the prescribed measurement. SPLOCS was second and clustered PCA presented the greatest differences (see Figure 19).

Even though our approach is the one that is closest (on average) to the prescribed measurements, there is a limitation due to the blending of the synthesized parts. This blending sometimes affects the mesh in a way that prevents it from achieving the exact prescribed effect for the editing. Figure 20 shows an example where the blending does not maintain the "Nose Height" of the synthesized nose as it deforms it through the blending process.

9.2 | Validation system

Being based on eigenvectors extracted from a data set, our approach has a strong tendency to generate realistic faces. Nevertheless, when the realism is of greater importance, we might want to validate how realistic these faces are. In creating new faces, it is always possible to disrupt facial features and their interrelations, leading to decreased realism. Studying the literature, we selected two categories of validation approaches. The first category corresponds to metrics. We selected some of the neoclassical canons of facial proportions[23] as well as the angles used most frequently in rhinoplasty surgeries[28] (Nasfrontal Angle, Facial Convexity, and Total Facial Convexity). The second type of validation considers the depth of the nose compared to the other features.[29] We validate that the nose is always upfront compared to other parts of the face (such as the chin, lips, and forehead).

For the metrics, we calculated their minimum and maximum values on a data set of 133 3D scans. We then used a second data set (358 3D scans) to verify if our ranges of values generalized. The initial percentage of the heads not fitting in the ranges was 15.5%. We then made slight adjustments to the initial ranges (less than 0.1 for the proportions and less than 5° for the angles), and the percentage of the heads not fitting in the ranges went from 15.5% to only 1.1%. While the remaining heads that still failed the test were real ones, it is important to mention that they were having very specific facial features such as a frown mark or open mouth, which explains why they failed the validation. Table 4 shows the final determined ranges for these metrics. Apart from the metrics, the nose depth is also a powerful strategy to determine the realism. All of the heads (from the data set of 133 scans and the data set of 358 scans) all passed this test.

We then created 124 new heads using our pipeline, all with randomly exaggerated features. In creating one such head, we select a random head from the data set, we randomly select one of the measurements (Section 6.1), and randomly set the value so that it is 1.5–2 times the maximal value observed in the data set. We ran these randomly edited heads through

TABLE 4 The ranges for the facial metrics

Metrics	Min	Max
$\|\|en-en\|\| \div \|\|en-ex\|\|$	0.95	1.71
$\|\|en-en\|\| \div \|\|al-al\|\|$	0.75	1.284
$\|\|n-sn\|\| \div \|\|sa-sba\|\|$	0.68	1.184
$\|\|tr-n\|\| \div \|\|n-sn\|\|$	0.92	1.366
$\|\|tr-n\|\| \div \|\|sn-gn\|\|$	0.64	1.086
$\|\|n-sn\|\| \div \|\|sn-gn\|\|$	0.5	0.97
$\|\|al-al\|\| \div (\|\| zy-zy\|\|/4)$	0.9	1.471
$\|\|ch-ch\|\| \div \|\|al-al\|\|$	1.12	1.8
TotalFacialConvexity	127°	158.6°
FacialConvexity	147.46°	178.5°
NasfrontalAngle	127°	167.28°

Note: Specific metrics and symbols are explained in the accompanying video.

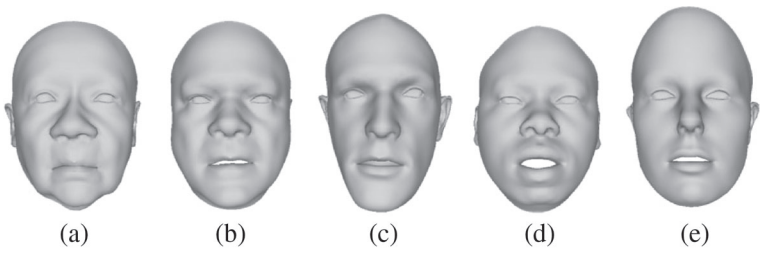


FIGURE 21 Some samples of the heads built using our pipeline which did not pass the validation test. The problematic metrics are as follows: (a) $||ch-ch|| \div ||al-al||$ and $||en-en|| \div ||en-ex||$, (b) $||en-en|| \div ||en-ex||$ and $||n-sn|| \div ||sa-sba||$, (c) $||en-en|| \div ||en-ex||$, (d) tip of the nose, (e) $||al-al|| \div (|| zy-zy||/4)$ and $||tr-n|| \div ||n-sn||$

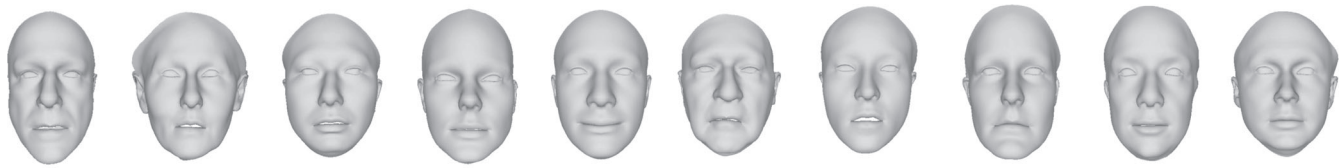


FIGURE 22 Some samples of the heads built using our pipeline which passed the validation test

our set of rules, and as a result, we had 15.3% of the heads which were not validated as realistic heads (we determine that a realistic head is within all of the ranges and also passes the nose depth test). Figure 21 shows some of the completely new created faces which did not pass the test. Conversely, Figure 22 shows faces which passed the test.

9.3 | Face decomposition

Our face segmentation was motivated by several facial animation artists with whom we worked, and who strongly prefer having control over the face patches in order to make sure they match the morphology of the face and muscle locations. This type of control is impossible to achieve with an automatic method, which is typically agnostic to the underlying anatomical structure. It is important to note that this manual way of selecting the regions is no more cumbersome than the current state-of-the-art methods. The state-of-the-art method of Tena et al.[8] requires a post-processing step to fix occasional artifacts in the segmentation method. Furthermore, as illustrated in Figure 9(b), segmentation boundaries

can occasionally occur across important semantic regions such as the eyes, leading to complications further down the pipeline.

9.4 | Data set

The quality of the input mesh data set is important to the reconstruction of good 3D face models. As many existing methods, we assume that the meshes share a common mesh topology. Mapping the raw 3D scans to a common base mesh is typically done by a surface mapping method.[30-32] We established this correspondence with R3DS WRAP.

Our data preparation pipeline involves few steps leading to an automatic surface mapping. Since the 3D scans usually have mesh topology problems such as non-manifold edges, singular vertices, and holes, the first step is to clean the scans. We then seek to automatically find a good frontal camera alignment. As we do not know where the face of the person is in the scan coordinate frame, we generate 42 cameras around the cleaned scan using a Fibonacci sphere algorithm. The frontal pose detector[33] reads the images from the 42 cameras and scores each detection. After identifying the camera with the best score, this camera is rotated every 18° to find a good camera roll and compensate for the error between the rotation and the detector's trained angle (detector is trained on upright faces). From this frontal camera, we will use automatic facial landmark identification to align the scan to the local coordinate space of the base mesh. We automatically identify a set of 68 facial landmarks utilizing the facial landmark predictor of dlib.[34] These landmarks are back-projected on the 3D scan meshes (as illustrated in Figure 23) and used to do a rigid alignment to the base mesh coordinate space. In this coordinate space, a frontal pose camera is already set up, providing a very good frontal pose. As the precision of the landmark position depends on the camera orientation, we rerun the facial landmark identification given the camera frontal pose in the base mesh coordinate space. Widely used facial landmark detectors, such as those of dlib,[34] do not provide landmarks for the ears. At this point, we rotate the camera 90° to left/right for detecting left/right ears' bounding boxes[35] and we place an initial guess for the ear landmarks[36] which we further correct by running an active appearance model[37] trained on AMI ear database. Finally, we use the landmarks and aligned scans as inputs to wrap a base mesh with 6014 vertices and 11,964 triangles around the high-resolution scans. We also re-project the textures from the scans on the wrapped base mesh. Our data set represent a diverse range of ethnicity and ages (Figure 24) which is in contrast to the Basel Face Model.[38]

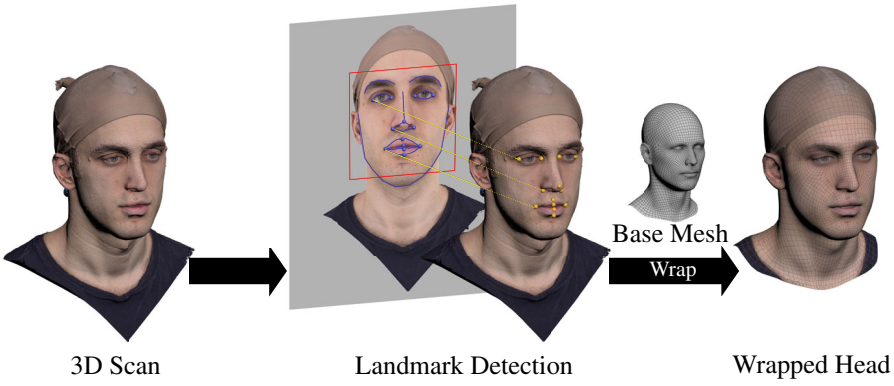


FIGURE 23 We automatically identify landmarks on the scan which will be used to establish the correspondence to the base mesh and create the final wrapped head

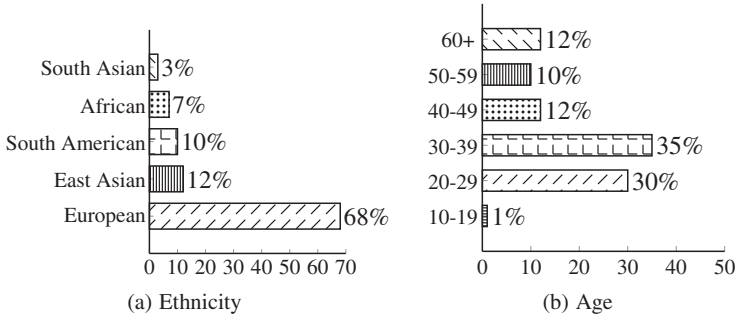


FIGURE 24 Racial and age distribution of our data set. It includes 135 3D scans consisting of 64 females and 71 males. (a) The racial distribution of these heads while (b) different age groups in the data set

10 | CONCLUSION

In this article, we designed a new local 3DMM used for face editing. We demonstrated the difficulty of locally editing the face with global 3DMMs; we thus segmented the face into five parts and combined the 3DMMs for each part into a single 3DMM by selecting the best eigenvectors through prediction error measurements. We then proposed the use of established anthropometric measurements as a basis for face editing. We mapped the anthropometric measurements to the 3DMM through a mapping matrix. We proposed a process to select the best set of anthropometric measurements, leading to improved reconstruction accuracy and the removal of conflicting measurements. From a list of 33 anthropometric measurements we surveyed from the literature, we identified 31 which lead to an improvement of the reconstruction and rejected 2 as they decreased the quality of the reconstruction. Note that the anthropometric measurement selection process would apply as well even if using a different 3DMM from the one proposed in this article, as well as when considering a different set of anthropometric measurements. We demonstrated this by applying our set of measurements to both SPLOCS[10] and clustered PCA.[8] This also demonstrated that our approach produces results superior to those of established methods proposing automatic segmentation and different ways to construct the eigenvector basis. We also presented different bits of experimental evidence to demonstrate the superiority of our approach, especially in terms of local control, as compared to the typical global 3DMM.

A limitation of our approach lies in the mapping matrices, which assume a linear relationship between anthropometric measurements and the eigenvector weights. An interesting avenue for future work would be to apply machine learning to identify nonlinear mappings.

Also, our measurements are based on distances between points on the surface. Future work could consider measurements based on the curvature over the face, such as measurements specifying the angle formed at the tip of the chin.

Although anthropometric measurements generate plausible facial geometric variations, they do not consider fine-scale or coarse-scale features. Regarding the fine-scale details, our approach does not model realistic variations of wrinkles, and that could be an interesting direction for future research. Regarding coarse-scale features, we could reconstruct a skull based on the anthropometric measurements, and then generate the facial mask based on an energy minimization of the skin thickness considering the skull and the measurements.

ACKNOWLEDGMENTS

This work was supported by Ubisoft Inc., the Mitacs Accelerate Program, and École de technologie supérieure.

ORCID

Donya Ghafourzadeh https://orcid.org/0000-0001-9781-936X

REFERENCES

1. Blanz V, Vetter T. A morphable model for the synthesis of 3D faces. Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '99. New York, NY: ACM Press/Addison-Wesley Publishing Company; 1999. p. 187–94. https://doi.org/10.1145/311535.311556.
2. Allen B, Curless B, Popović Z. The space of human body shapes: reconstruction and parameterization from range scans. Proceedings of the ACM SIGGRAPH 2003 Papers. SIGGRAPH '03. New York, NY: Association for Computing Machinery; 2003. p. 587–94. https://doi.org/10.1145/1201775.882311.
3. Cao C, Weng Y, Zhou S, Tong Y, Zhou K. FaceWarehouse: a 3D facial expression database for visual computing. IEEE Trans Vis Comput Graph. 2014;20(03):413–25.
4. Booth J, Roussos A, Zafeiriou S, Ponniah A, Dunaway D. A 3D morphable model learnt from 10,000 faces. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV; 2016. p. 5543–52.
5. Li T, Bolkart T, Black MJ, Li H, Romero J. Learning a model of facial shape and expression from 4D scans. ACM Trans Graph. 2017;36(6):1–17. https://doi.org/10.1145/3130800.3130813.
6. Booth J, Roussos A, Ponniah A, Dunaway D, Zafeiriou S. Large scale 3D morphable models. Int J Comput Vision. 2018;126(2–4):233–54. https://doi.org/10.1007/s11263-017-1009-7.
7. Lüthi M, Gerig T, Jud C, Vetter T. Gaussian process morphable models. IEEE Trans Pattern Anal Mach Intell. 2018;40:1860–73.
8. Tena JR, De la Torre F, Matthews I. Interactive region-based linear 3D face models. Proceedings of the ACM SIGGRAPH 2011 Papers. SIGGRAPH '11. New York, NY: Association for Computing Machinery; 2011. https://doi.org/10.1145/1964921.1964971.
9. Chi J, Gao S, Zhang C. Interactive facial expression editing based on spatio-temporal coherency. Vis Comput. 2017;33(6–8):981–91. https://doi.org/10.1007/s00371-017-1387-4.

10. Neumann T, Varanasi K, Wenger S, Wacker M, Magnor M, Theobalt C. Sparse localized deformation components. ACM Trans Graph. 2013;32(6):1–10. https://doi.org/10.1145/2508363.2508417.
11. Cao C, Chai M, Woodford O, Luo L. Stabilized real-time face tracking via a learned dynamic rigidity prior. ACM Trans Graph. 2018;37(6):1–11. https://doi.org/10.1145/3272127.3275093.
12. Cao C, Bradley D, Zhou K, Beeler T. Real-time high-fidelity facial performance capture. ACM Trans Graph. 2015;34(4):1–9. https://doi.org/10.1145/2766943.
13. Garrido P, Zollhöfer M, Casas D, Valgaerts L, Varanasi K, Pérez P, et al. Reconstruction of personalized 3D face rigs from monocular video. ACM Trans Graph. 2016;35(3):1–15. https://doi.org/10.1145/2890493.
14. Shi F, Wu HT, Tong X, Chai J. Automatic acquisition of high-fidelity facial performances using monocular videos. ACM Trans Graph. 2014;33(6):1–13. https://doi.org/10.1145/2661229.2661290.
15. Ichim AE, Kadleček P, Kavan L, Pauly M. Phace: physics-based face modeling and animation. ACM Trans Graph. 2017;36(4):1–14. https://doi.org/10.1145/3072959.3073664.
16. Wu C, Bradley D, Gross M, Beeler T. An anatomically-constrained local deformation model for monocular face capture. ACM Trans Graph. 2016;35(4):1–12. https://doi.org/10.1145/2897824.2925882.
17. Streuber S, Quiros-Ramirez MA, Hill MQ, Hahn CA, Zuffi S, O'Toole A, et al. Body talk: crowdshaping realistic 3D avatars with words. ACM Trans Graph. 2016;35(4):1–14. https://doi.org/10.1145/2897824.2925981.
18. Ghafourzadeh D, Rahgoshay C, Fallahdoust S, Beauchamp A, Aubame A, Popa T, et al. Part-based 3D face morphable model with anthropometric local control. Toronto, ON: Proceedings of Graphics Interface 2020. Canadian Human-Computer Communications Society; 2020. p. 7–16.
19. Sorkine O, Cohen-Or D, Lipman Y, Alexa M, Rössl C, Seidel HP. Laplacian surface editing. Proceedings of the 2004 Eurographics/ ACM SIGGRAPH Symposium on Geometry Processing. SGP '04. New York, NY: ACM; 2004. p. 175–84.
20. Sorkine O, Alexa M. As-rigid-as-possible surface modeling. Proceedings of the 5th Eurographics Symposium on Geometry Processing. SGP '07. Goslar, DEU: Eurographics Association; 2007. p. 109–16.
21. Etöz A, Ercan I. Anthropometric analysis of the nose. Handbook of anthropometry: physical measures of human form in health and disease. New York, NY: Springer; 2012. p:919–26.
22. Farkas LG. Anthropometry of the head and face. New York, NY: Raven Pr; 1994.
23. Farkas LG, Katic MJ, Forrest CR. International anthropometric study of facial morphology in various ethnic groups/races. J Craniofacial Surg. 2005;16:615–46.
24. Lacko D, Huysmans T, Parizel PM, De Bruyne G, Verwulgen S, Van Hulle MM, et al. Evaluation of an anthropometric shape model of the human scalp. Appl Ergon. 2015;48:70–85.
25. Lee JH, Shin SJ, Istook C. Analysis of human head shapes in the United States. Int J Hum Ecol. 2006;01:7.
26. Ramanathan N, Chellappa R. Modeling age progression in young faces. Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition—Vol. 1. CVPR '06. New York, NY: IEEE Computer Society; 2006. p. 387–94. https://doi.org/10.1109/CVPR.2006.187.
27. Zhuang Z, Landsittel D, Benson S, Roberge R, Shaffer R. Facial anthropometric differences among gender, ethnicity, and age groups. Ann Occupat Hygiene. 2010;54(3):391–402.
28. Park CW, Lee MJ, Jung YI. Photogrammetric facial analysis of attractive celebrities using the glabella for planning rhinoplasty and analyzing surgical outcomes. Arch Aesthetic Plast Surg. 2018;24:105–10.
29. Majeed R, Beiji Z, Hatem H. Nose tip detection in 3D face image based on maximum intensity algorithm. Int J Multimed Ubiquit Eng. 2015;10:373–82.
30. Ramachandran S, Ghafourzadeh D, de Lasa M, Popa T, Paquette E. Joint planar parameterization of segmented parts and cage deformation for dense correspondence. Comput Graph. 2018;74:202–12.
31. Zell E, Botsch M. ElastiFace: matching and blending textured faces. Proceedings of the Symposium on Non-Photorealistic Animation and Rendering. NPAR '13. New York, NY: ACM; 2013. p. 15–24. https://doi.org/10.1145/2486042.2486045.
32. Amberg B, Romdhani S, Vetter T. Optimal step nonrigid ICP algorithms for surface registration. Minneapolis, MN: Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition. IEEE; 2007. p. 1–8.
33. King DE. Dlib-ml: a machine learning toolkit. J Mach Learn Res. 2009;10(Jul):1755–8.
34. Kazemi V, Sullivan J. One millisecond face alignment with an ensemble of regression trees. Columbus, OH: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition; 2014.
35. Castrillón-Santana M, Lorenzo-Navarro J, Hernández-Sosa D. An study on ear detection and its applications to face detection. Proceedings of the 14th International Conference on Advances in Artificial Intelligence: Spanish Association for Artificial Intelligence. CAEPIA'11. New York, NY: Springer-Verlag; 2011. p. 313–22.
36. Pflug A, Busch C. Ear biometrics: a survey of detection, feature extraction and recognition methods. IET Biometr. 2012;1:114–29.
37. Alabort-i-Medina J, Zafeiriou S. A unified framework for compositional fitting of active appearance models. Int J Comput Vis. 2017;121:26–64.
38. Paysan P, Knothe R, Amberg B, Romdhani S, Vetter T. A 3D face model for pose and illumination invariant face recognition. Proceedings of the 2009 6th IEEE International Conference on Advanced Video and Signal Based Surveillance. AVSS '09. Washington, DC: IEEE Computer Society; 2009. p. 296–301.

AUTHOR BIOGRAPHIES

Donya Ghafourzadeh holds a Ph.D. from École de technologie supérieure (ÉTS), Montréal, Canada under the supervision of Professor Eric Paquette. Her dissertation was conducted in collaboration with Autodesk and Ubisoft Montréal concentrating on characters animation and facial modeling. She received her M.Sc. degree in Advanced Computer Graphics from Linköping University, Sweden.

Sahel Fallahdoust is a generalist programmer at Helix, Ubisoft Montréal for almost 5 years now. Apart from tools and shader integrations at Helix, she also has been actively collaborating with La Forge in R&D projects. She received her Bachelor in Computer Science from AUT and a Master in Computer Graphics at École de technologie supérieure (ÉTS), specialized in character animation.

Cyrus Rahgoshay is a multicultural globe-trotter with a passion for Computer graphics and animation. After receiving his M.Sc. in Computer Science and physically based animation from McGill University in 2012, he began his career as an R&D Software Developer at CMLabs Simulations where he found his passion for Game engines. In 2014, Cyrus joined Ubisoft as Software Developer where he builds new pipeline and tools for artists to replicate and use in Game assets in content creation tools and they, in turn, use the tools to create 2D and 3D contents for numerous AAA Games such as Assassin's Creed, Rainbow Six Siege, Watch Dogs, and much more in surprising and unexpected ways.

Andre Beauchamp is an R&D developer at La Forge, R&D center for the video game company Ubisoft Entertainment Inc. located in Montreal, Canada. His past work at Ubisoft touches all aspects of the video game development, but mostly R&D about 3D rendering, physics, simulations and procedural generation of all sorts including textures and 3D meshes. M. Beauchamp had graduated as an engineer (B.Eng.) and received his Master degree in Applied Science at Ecole Polytechnique de Montréal, Canada.

Adeline Aubame has around 15 years of experience in the Visual Effect and Video Game industry. She graduated from Laval university with a Bachelor in Graphic Communication and then did one intensive year at the Nad Center in the Animation and visual Effects for Television and Cinema program. She occupied various position in the gaming and VFX industry such as 3D Generalist, Lead Animator, Animation TD, VFX TD, CG Sup, and today, as the Technology Operations Director of Helix, Ubisoft Montréal.

Tiberiu Popa is an Associate Professor at Concordia University, Montréal. He completed his Bachelor of Mathematics in 2001 and Master of Mathematics in 2003, both at the University of Waterloo in Canada. In 2010, Tiberiu obtained a PhD from the University of British Columbia in Canada that received the Alain Fournier annual thesis award, and then he was a post-doctoral fellow and later a senior researcher at ETH Zurich until June 2013. Tiberiu's main research interests are in digital geometry processing, spatial-temporal surface acquisition, free viewpoint video.

Eric Paquette is a full professor at École de technologie supérieure (ÉTS), Montréal, Canada. He teaches Computer Graphics in the Software & IT Engineering Department and does his research at the Multimedia Lab. His research interests include geometry processing, character modeling and animation, texture synthesis, fluid simulation, and simulation of natural phenomena. Professor Paquette received his B.Sc. in Computer Science from Université de Sherbrooke, Canada, his M.Sc. degree in Computer Science from Université de Montréal, Canada, and his Ph.D. degree in Computer Science from Université de Montréal, Canada and Université Joseph Fourier, France.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of this article.

How to cite this article: Ghafourzadeh D, Fallahdoust S, Rahgoshay C, Beauchamp A, Aubame A, Popa T, Paquette E. Local control editing paradigms for part-based 3D face morphable models. Comput Anim Virtual Worlds. 2021;32:e2028. https://doi.org/10.1002/cav.2028