ImFace: A Nonlinear 3D Morphable Face Model with Implicit Neural Representations

Supplementary material

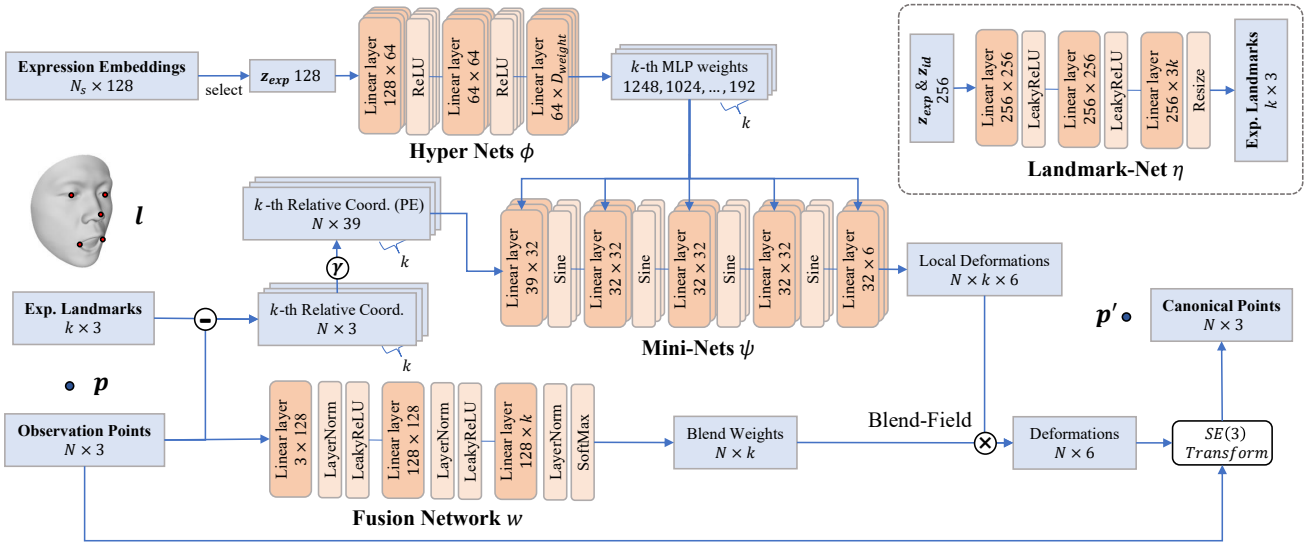


Figure 1. Detailed architecture of Expression Mini-Nets block.

A. Detailed Network Architecture

The detailed architectures of Expression Mini-Nets, Identity Mini-Nets, and Template Mini-Nets are shown in Fig. 1, Fig. 3 and Fig. 4 respectively, where N_s refers to the number of total scans, N_{id} denotes the number of identities, and PE indicates positional encoding.

All the networks are fully implemented by MLPs. To achieve better performance for high-frequency clues, we encode the relative coordinates with respect to k landmarks by sinusoidal positional encoding γ [6], written as $\gamma(p) = (\sin(2^0\pi p), \cos(2^0\pi p), \cdots, \sin(2^{L-1}\pi p), \cos(2^{L-1}\pi p))$, where $L = 6$, $k = 5$ in our experiments. Besides, sine activations are exploited in every Mini-Net ψ_n and the parameters are initialized as in [9]. During training, the parameters of k Mini-Nets are generated by k corresponding Hyper Nets for a more expressive latent space, which is a common technical operation in recent INRs studies [2, 8].

The trade-off parameters $\lambda_1, \cdots, \lambda_7$ to train the networks are set to $3e3$, $1e2$, $5e1$, $1e6$, $1e2$, $1e2$, and $1e2$ respectively.

B. Preprocessing

To enable INRs to work with non-watertight 3D faces, we present an effective preprocessing pipeline as briefly described in Sec. 3.5, so that facial geometry and correspondence can be learned as exquisitely as on watertight objects.

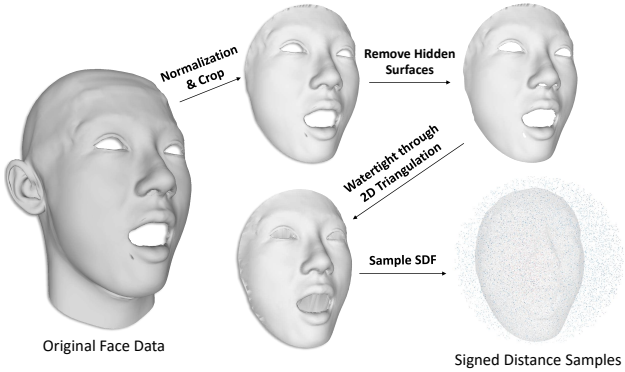


Figure 2. Illustration of the preprocessing pipeline.

We first determine the domain of definition $\{(x, y, z) \in \mathbb{R}^3\}$ of our implicit function $f : \mathbb{R}^3 \mapsto \mathbb{R}$. Specifically, the coordinate origin is set at the point 4 cm behind the nose tip. This setting helps to balance the number of positive and negative SDF samples, which is crucial to facilitate INRs network convergence. To cover most of facial geometries while cut away unnecessary regions, a sphere S with a radius of 10 cm centered on the coordinate origin is defined as the sampling area. However, it is not an intuitive task to determine whether a point in S is "inside" or "outside" of a 3D facial surface, mainly because facial surface may contain multiple openings such as the mouth and eyes, as well as complex geometric structures in the nasal or oral cavity,

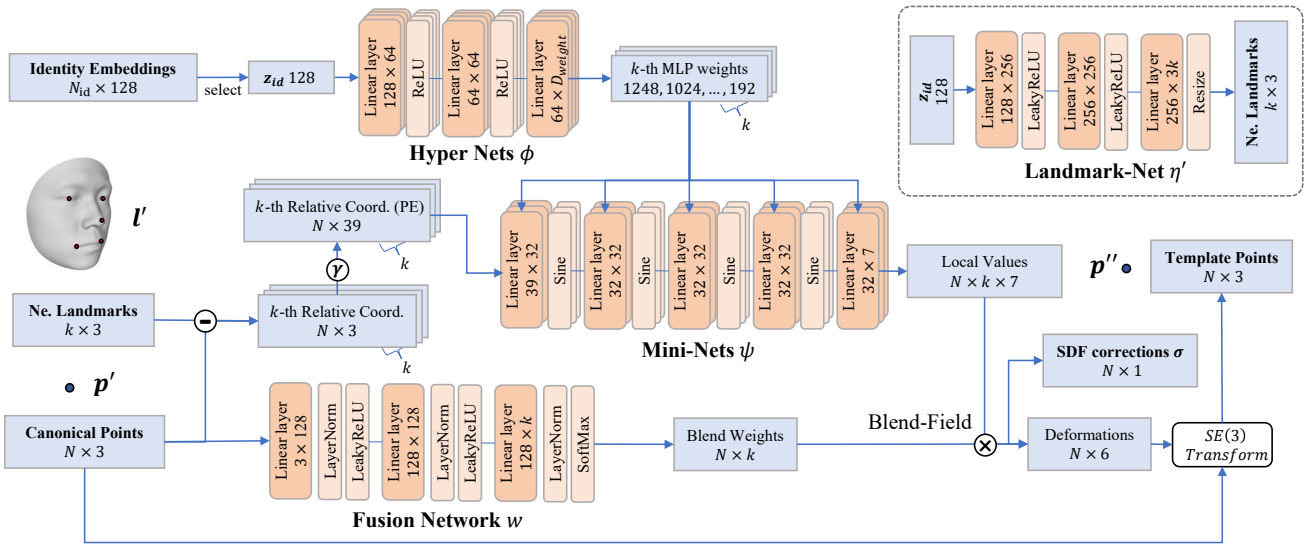


Figure 3. Detailed architecture of Identity Mini-Nets block. It additionally predicts a correction term to cope with possible non-existent correspondences.

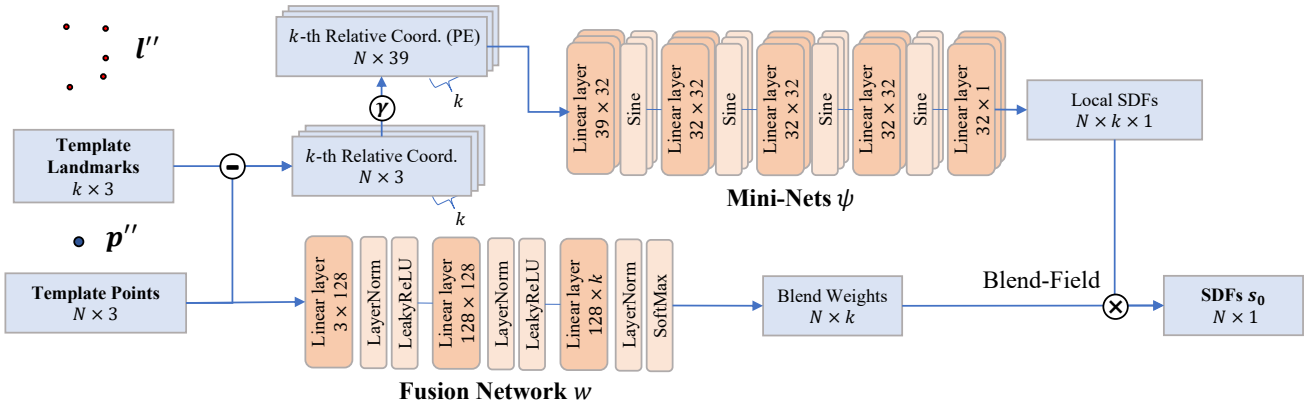


Figure 4. Detailed architecture of Template Mini-Nets block. It does not require a conditional embedding since the template face is shared across all faces.

depending on the acquisition conditions. The proposed pre-processing pipeline aims to address the issue above.

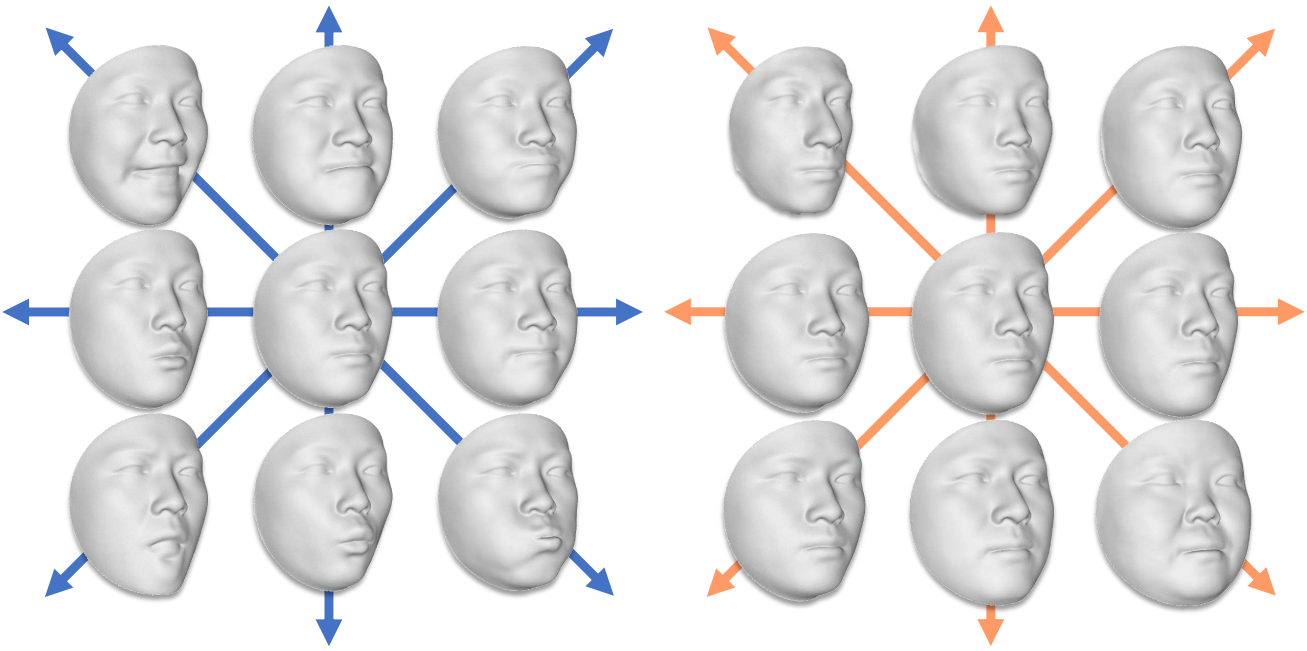
Our key observation is that, if a 3D surface C satisfies the following property, it can be oriented easily. Without loss of generality, we consider an infinite continuous surface C defined in 3D space $\Omega : \{(x, y, z) \in \mathbb{R}^3\}$.

Definition 1. *Assuming C is defined by an implicit function $f(x, y, z) = 0$, if z is an injective function of (x, y), we call C is injective at z.*

Property 1. *If C is injective at z, Ω is divided into two and only two spaces Ω_1, Ω_2, that for any $(x, y, z) \in \Omega_1$, a ray from (x, y, z) along the positive z-axis intersects with C for only once, and for any $(x, y, z) \in \Omega_2$ it does not intersect with C.*

Proof. It is equivalent to prove that for any $(x, y, z) \in \Omega$, the ray starting from it along the positive z-axis cannot intersect with C at more than one point. By reductio, if a ray l from $(x_0, y_0, z_0) \in \Omega$ intersects with C at multiple different points $\{(x_1, y_1, z_1), \cdots, (x_n, y_n, z_n)\}(n \geq 2)$, we have $(x_1, y_1) = \cdots = (x_n, y_n) = (x_0, y_0)$ but $z_1 \neq \cdots \neq z_n$, which conflicts with the injective precondition of z.

Based on the property above, we acknowledge that if a facial surface satisfies the property in ***Definition 1***, then two separate 3D regions can be clearly determined and the inside and outside space can be set manually, so that SDF can be further defined. In general, human faces are approximately injective at the frontal direction. To make it strictly satisfy ***Property 1***, for any face mesh (V, F) we generate new triangles F' by performing the Delaunay Triangulation Algorithm [3] on x-y coordinate and construct a new



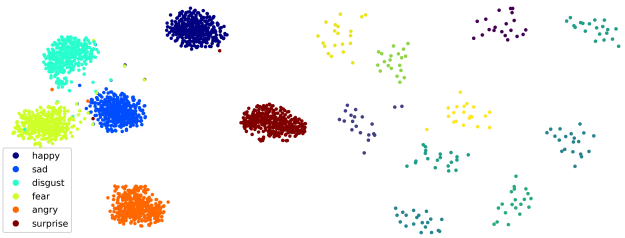
(a) Expression Componets (b) Identity Componets

Figure 5. Visualization of the expression and identity principal components of ImFace on FaceScape [11].

mesh (V, F'), thus any straight line parallel to z-axis only intersects with one triangle in F'. It holds because a Triangulation Algorithm covers the convex hull and does not lead to overlaps between 2D triangles, which makes any x-y coordinate have a unique triangle corresponding to it.

Considering that directly performing 2D triangulation makes the points on hidden surfaces (*e.g.* the inner surface of nasal cavity) interlace with the ones on frontal surface, leading to unreasonable triangles, the Ray-Triangle Intersection Algorithm [7] is thus iteratively executed to remove the hidden surfaces before triangulation. Specifically, a vertex is marked if a ray from it along the positive z-axis intersects with more than one triangle in F, and the triangles in F which have marked vertices are removed. In this way, a pseudo watertight face mesh can be established without much loss of accuracy, which divides the sampling space into two separate parts clearly.

Given a preprocessed face mesh, the sign of any query point in the sampling sphere can be determined by whether a ray from it to positive z-axis intersects with the face mesh, as in ***Property 1***. In order to accelerate the calculation procedure, we make use of the distance vectors calculated via distance transform and determine the sign of a query point by the angle between its distance vector to the nearest surface and the positive direction of z-axis , which is equivalent to the above-mentioned ray intersection checks. The whole preprocessing pipeline is presented in Fig. 2.



(a) (b)

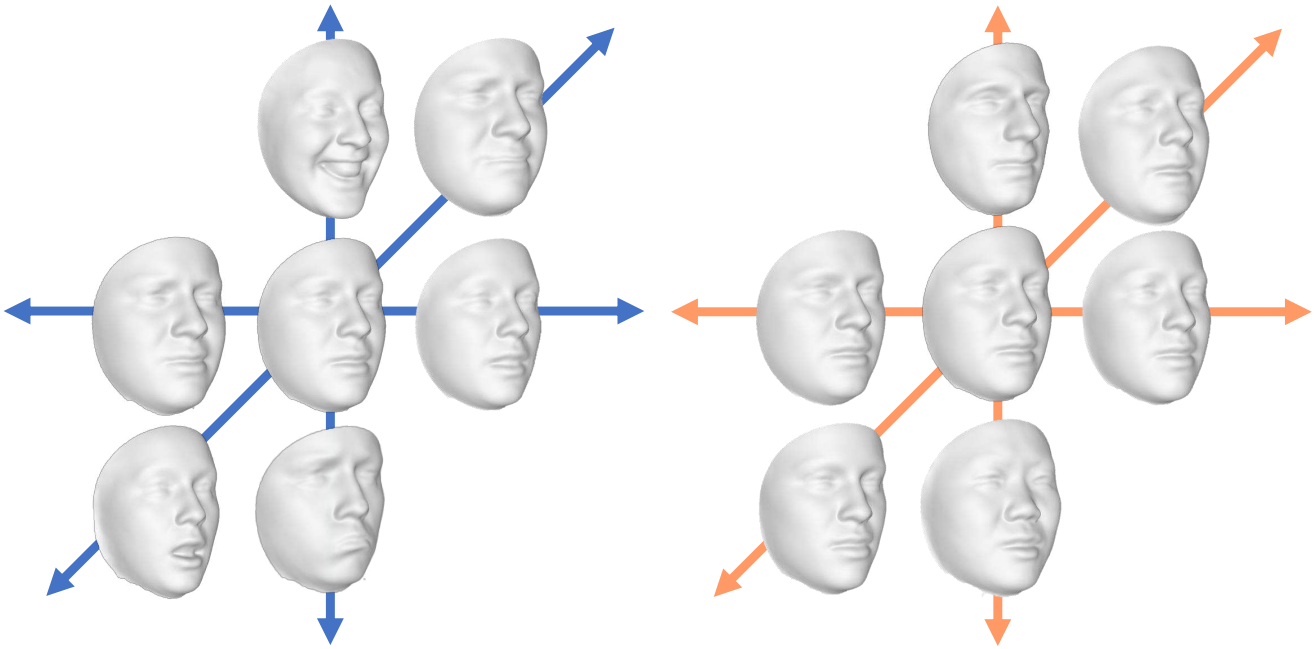
Figure 6. Visualizing the distributions of high-dimensional expression and identity embeddings with t-SNE. **(a)** the embedding distribution of 6 typical expressions from the training set. **(b)** the identity embedding distribution from the test set.

C. Experiments

To have a better insight into the proposed ImFace morphable model, we provide more evaluation results in this supplementary material.

C.1. Face Variation Visualization

We apply Principal Components Analysis (PCA) on the learned expression and identity embeddings to visualize model variations, as shown in Fig. 5. The standard deviations in terms of expression and identity are set to ± 3 and ± 30 respectively. In particular, four expression principal



(a) Expression Componets (b) Identity Componets

Figure 7. Visualization of the expression and identity principal components on BU-3DFE [13].

Figure 8. Cross-dataset reconstruction (trained on FaceScape, tested on BU-3DFE).

components are visualized in Fig. 5 (a). Despite great expression changes, the faces maintain a consistent identity. Besides, by learning expression components from thousands of unique embeddings, vivid expressions can be produced by ImFace. In Fig. 5 (b), we can observe a similar phenomenon on the learned identity components that the facial expressions remain stable when identity varies. The experimental results indicate that a good distanglement between expression and identity is achieved, which is crucial to generating novel faces by reweighting the singular values.

C.2. High-dimensional Embedding Visualization

To validate this point, we visualize the learned high-dimensional expression embeddings from 2,130 training scans with 6 typical expressions by t-SNE [10], as Fig. 6 (a) shows. It can be seen that our network is capable of unsupervisedly distinguishing different expression types only by learning from expression-related shape morphs, which indicates its superior ability in expression modeling. Furthermore, we visualize the identity embeddings from the test set in Fig. 6 (b), which involves 200 face scans from 10 persons. Visually inspected, our model successfully captures different identity features even under various complicated expressions.

C.3. More Results on FaceScape

In Fig. 9 and Fig. 10, we present more comparison with i3DMM [12], FLAME [5], FaceScape [11], and ground-truth faces, where a common color-coded distance (fit-to-scan) is used to indicate the reconstruction errors. As can be seen, faces are reconstructed more accurately by ImFace than the counterparts.

C.4. Results on BU-3DFE

We additionally preprocess the BU-3DFE [13] database and train ImFace on it. Fig. 7 displays the expression and identity principal components achieved on BU-3DFE. As can be seen, although this dataset contains some pose variations, ImFace still captures facial geometry faithfully, which validates its generality. Further, we give cross-dataset results in Fig. 8, which shows that our model well generalizes to another dataset.

C.5. Applications

ImFace is a general face representation established upon prior distributions of facial expression and identity morphs, and it can thus be applied to various down-stream applications. In Fig. 11, we provide expression editing results achieved on the FaceScape test set. The faces in the first row are the real ones providing source identities with neutral expressions, while the rest are generated by the proposed model. Our model is able to edit facial expression by simply changing their expression embeddings. The vivid 3D faces generated clearly validate the powerful representation ability of ImFace.

D. Visualization Techniques

We use Marching Cubes [4] to reconstruct facial surfaces from the signed distance field, where the voxel resolution is set to 256^3. All the meshes are rendered by Pyrender [1].

References

[1] https://github.com/mmatl/pyrender. 5

[2] Yu Deng, Jiaolong Yang, and Xin Tong. Deformed implicit field: Modeling 3d shapes with learned dense correspondence. In *CVPR*, 2021. 1

[3] Der-Tsai Lee and Bruce J Schachter. Two algorithms for constructing a delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242, 1980. 2

[4] Thomas Lewiner, Hélio Lopes, Antônio Wilson Vieira, and Geovan Tavares. Efficient implementation of marching cubes' cases with topological guarantees. *Journal of graphics tools*, 8(2):1–15, 2003. 5

[5] Tianye Li, Timo Bolkart, Michael J Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4d scans. *ACM TOG*, 36(6):194–1, 2017. 4, 6, 7

[6] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1

[7] Tomas Möller and Ben Trumbore. Fast, minimum storage ray-triangle intersection. *Journal of graphics tools*, 2(1):21–28, 1997. 3

[8] Vincent Sitzmann, Eric R. Chan, Richard Tucker, Noah Snavely, and Gordon Wetzstein. Metasdf: Meta-learning signed distance functions. In *NeurIPS*, 2020. 1

[9] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *NeurIPS*, 2020. 1

[10] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. 4

[11] 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 *CVPR*, 2020. 3, 4, 6, 7

[12] Tarun Yenamandra, Ayush Tewari, Florian Bernard, Hans-Peter Seidel, Mohamed Elgharib, Daniel Cremers, and Christian Theobalt. i3dmm: Deep implicit 3d morphable model of human heads. In *CVPR*, 2021. 4, 6, 7

[13] Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang, and Matthew J Rosato. A 3d facial expression database for facial behavior research. In *FGR*, 2006. 4

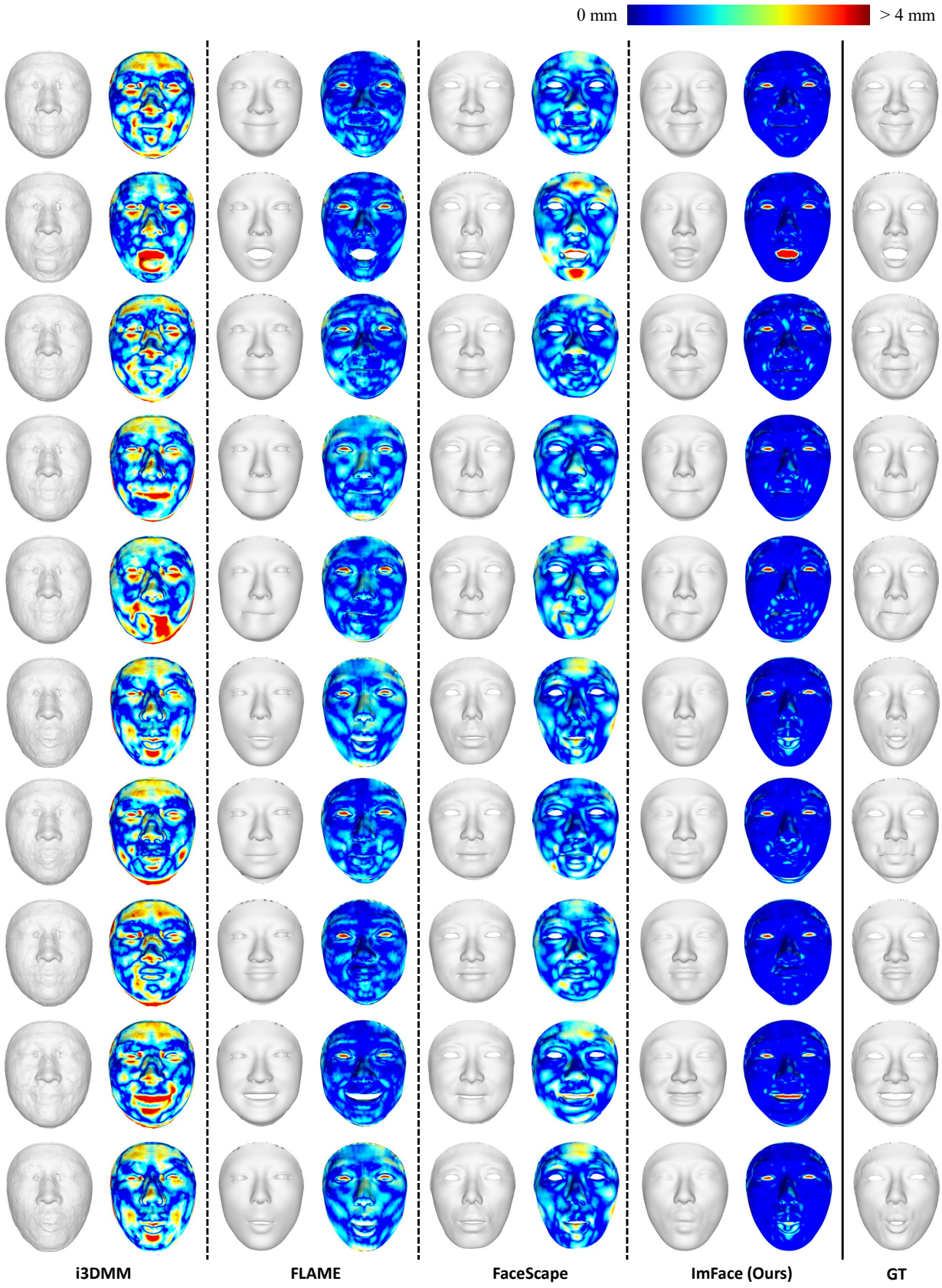


Figure 9. More comparison with i3DMM [12], FLAME [5], FaceScape [11], and ground-truth faces.

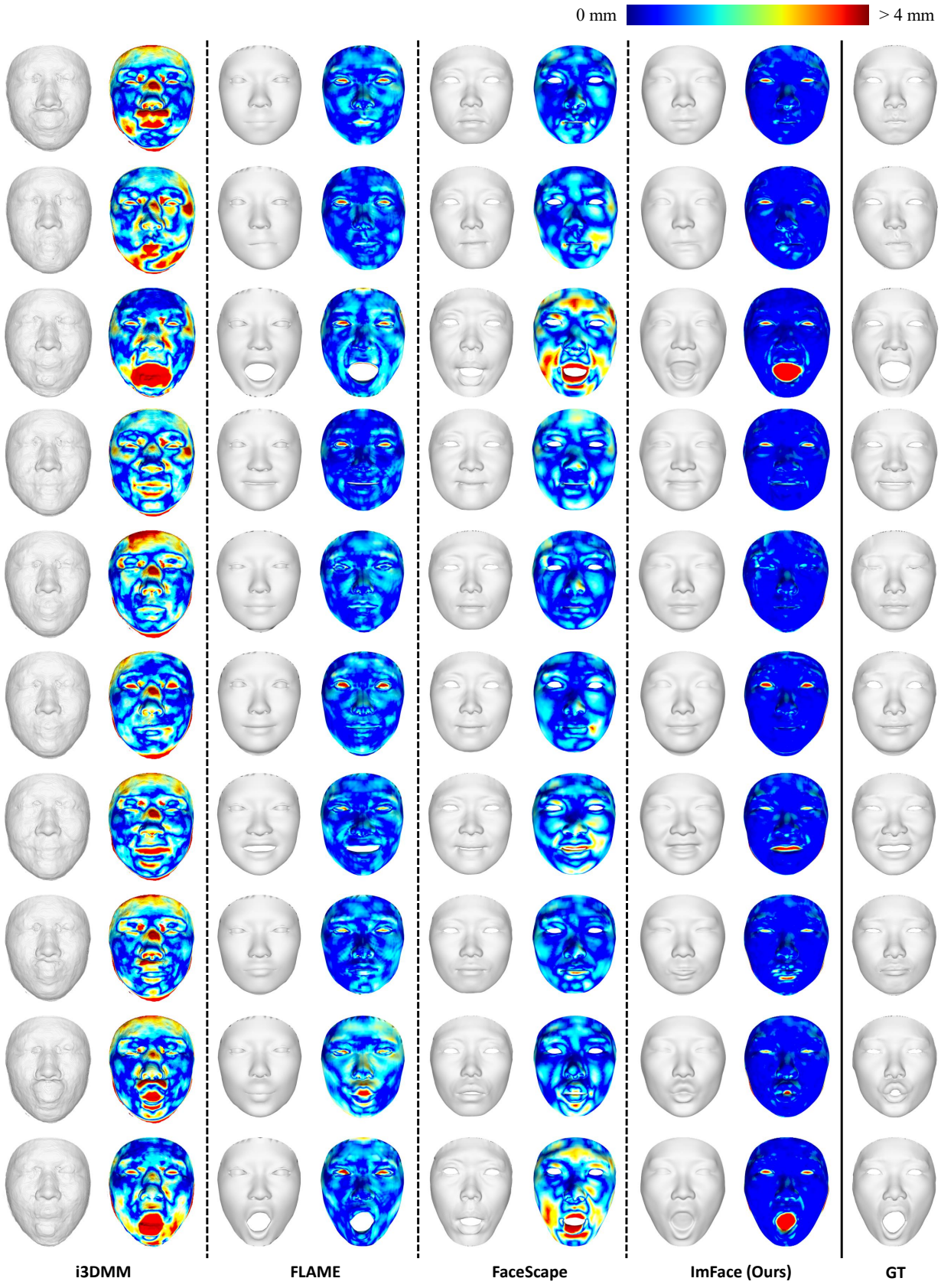


Figure 10. More comparison with i3DMM [12], FLAME [5], FaceScape [11], and ground-truth faces.

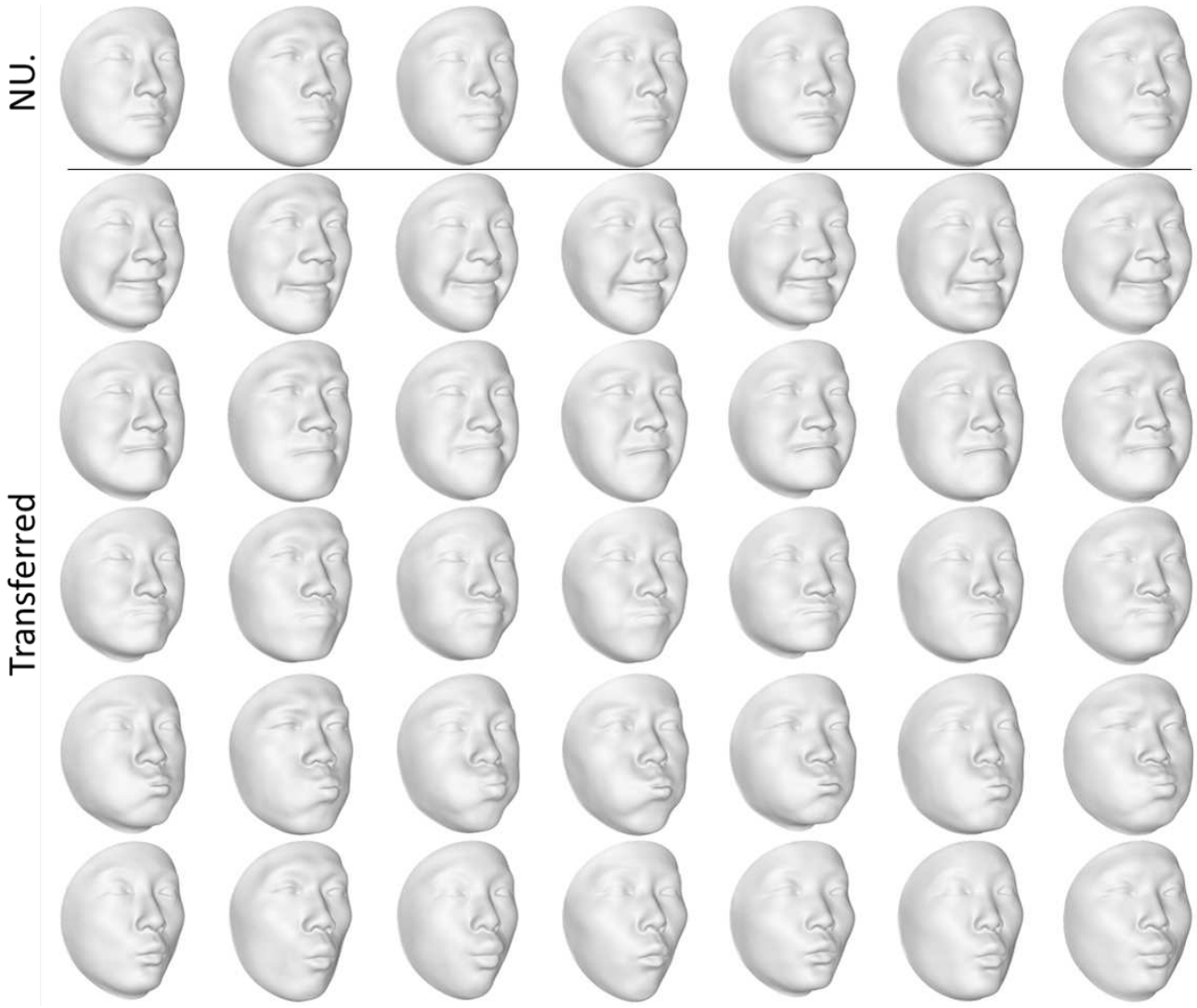


Figure 11. Expression editing results on the test set. The faces in the first row provide source identities, while the rest are generated by the proposed model. Our model is able to edit facial expression by exchanging embeddings.