Synthesizing Human Faces using Latent Space Factorization and Local Weights

Minyoung Kim and Young J. Kim*

Ewha Womans University, Seoul, 03760, South Korea
minyoung.mia.k@ewhain.net, kimy@ewha.ac.kr

Abstract. We propose a 3D face generative model with local weights to increase the model's variations and expressiveness. The proposed model allows partial manipulation of the face while still learning the whole face mesh. For this purpose, we address an effective way to extract local facial features from the entire data and explore a way to manipulate them during a holistic generation. First, we factorize the latent space of the whole face to the subspace indicating different parts of the face. In addition, local weights generated by non-negative matrix factorization are applied to the factorized latent space so that the decomposed part space is semantically meaningful. We experiment with our model and observe that effective facial part manipulation is possible, and that the model's expressiveness is improved.

Keywords: Face synthesis, Generative models, Learning-based approach

1 Introduction

Various methods have been studied to develop three-dimensional(3D) geometric models to generate human faces and related research using deep learning is being actively pursued. However, most existing works are focused on a holistic generative approach to generate all parts at once and lack part details and manipulation. Previous part-based generative models exploit explicit segmentation data or labels for training their model or use several part decoders[1, 2]. However, existing 3D facial mesh datasets barely have pre-segmented data.

Thus, we investigate an effective way to extract or present localized features from the whole data. Toward this goal, mesh segmentation might be one of the possible solutions. However, since human faces are often smooth, it is a challenge to segment the facial mesh explicitly. To bypass this, we exploit a generative approach that does not require additional segmentation data and makes the whole learning model simple. Furthermore, we explore a way for part control while exploiting holistic generation by learning localized features. In this paper, we propose a locally weighted 3D face generative model. Our approach can generate a rich variety of 3D face models beyond the training data using part manipulation with latent factorization. Latent space factorization enables manipulation of the local part of the face, and local weights make decomposed

part spaces more semantically meaningful without additional segmentation labels. With a part-based representation of the data, our model is simpler and more straightforward than others and does not require any semantic segmentation labels. As a basis model, we leverage Ranjan et al. [3]'s autoencoder with latent space factorization and apply local weights that partially influence the model during training. We also evaluate the performance of the proposed model in terms of part modification, part combination, and ablation tests to show the effect of each model component on the results.

Our main contributions are: (1) Locally weighted generative autoencoder for generating a whole human face geometric model; (2) End-to-end learning to learn local features without explicit facial feature segmentation data; (3) Experimentation and demonstration of the proposed model's performance in terms of generation and part manipulation. The majority of the materials contained in this paper are based on the same author's dissertation [4].

2 Related Work

There exist attempts to generate a new face with face segmentation and a local model to increase the model's expressiveness and achieve fine-scale modeling. Blanz and Vetter [5] demonstrated region-based modeling with 3D face morphable models (3DMMs) by manually dividing the face. Tena et al. [6] presented region-based linear face modeling with automatic segmentation by clustering.

CompoNet [1] presented a part-based generative neural network for shapes. They proved that the part-based model encourages the generator to create new data unseen in the training set. Dubrovina et al. [2] proposed decomposer-composer network performing meaningful part manipulation and high-fidelity 3D shape generation. They used projection matrices to split full object encodings into part encodings and represent them as fully connected layers. To composite each part, both [1] and [2] compute per-part affine transformation. Since we pursue a holistic generation approach, our model does not computes affine transformation to combinate each part of the data.

3 Locally Weighted Autoencoder

We choose to represent 3D faces with triangular mesh due to its efficiency. Among previous approaches for applying mesh convolution operation, Ranjan et al. [3] proposed CoMA employing fast Chebyshev filters [7] with a novel mesh pooling method. More details are referred to [3]. Although their work has shown a decent performance of reconstructing 3D faces, we take one step further to improve generation ability and controllability by using per-part manipulation. Utilizing the basic generation ability of Ranjan et al [3]'s model, we added two new methods: latent factorization and local weights.

Our model is based on the autoencoder [3] consisting of an encoder, projection, and a decoder, as illustrated in Fig. 1. The encoder and decoder learn how to compress and decompress the data, respectively. In between them, the

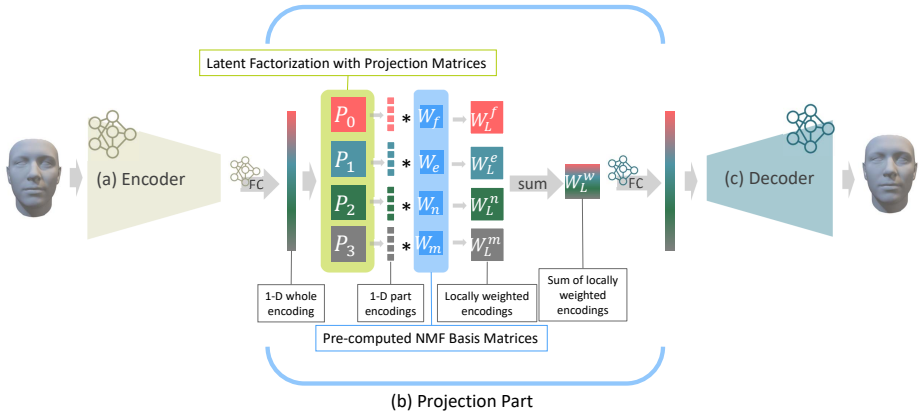


Fig. 1: Locally weighted autoencoder architecture

projection part factorizes the latent space into the subspace and applies local weights to make the subspace semantically meaningful. More details on local weights and latent space manipulation will be explained in the following section.

3.1 Pre-Computed Local Weights from NMF

We use a part-based representation to extract the local part structure without segmented data or labels. The representation is used as weights, which have each vertex's influence on each divided facial part. To make a part-based representation of the whole data, we employ the non-negative matrix factorization(NMF), which is a robust feature factorization method to represent data as part-based ones. This method finds a low-rank approximation of a matrix V, where $V \approx WH$, when V, W, and H do not have non-negative values. Given a feature matrix, V, W is a basis matrix that contains basis elements of V, and H is a latent representation matrix. We call the matrix W local weights. To express local features more efficiently, we applied sparse NMF [8] enforcing sparsity on the column of H. This could improve the local separation of features [9]. We compute this with a sparsity constraint value of 7.5. The computed local weights serve as the influence of each vertex on a specific area. We expect that local weights would make the part encodings more semantically meaningful. Fig. 2 shows the visualization of the local weights. The bright area shows how much each vertex influences the facial area.

Fig. 2: Pre-computed sparse NMF's basis matrix

3.2 Latent Space Manipulation

Projection Matrix Layer Our encoder takes a whole shape as input and compresses to a low-dimensional representation, i.e., a latent vector. This encoding reflects the whole shape structure. When we factorize the whole encoding, we can generate part encodings corresponding to the shape structure of the part. Thereby, we disentangle different semantic part encodings from the encoding of the whole shapes. We then perform part-level shape manipulation. Similarly to [2], we use learnable projection matrices to transform the whole part encoding from the global latent space to the localized basis matrix space. We define part-specific projection matrices, where K is the number of semantic parts. Passing through the matrices, the whole part encodings from the encoder are divided into semantic part encodings.

For embedding parts, we implement projection matrices represented as K fully connected layers without biases and with the latent dimension size of $Z \times Z$. The input of the projection layers is a whole face encoding produced by the encoder, and their outputs are K part encodings. The K part encodings can be split unpredictably and have arbitrary meanings. To make them more semantically meaningful, we apply pre-computed local weights.

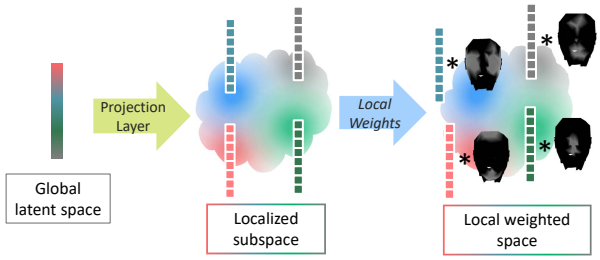


Fig. 3: Illustration of the projection part

Applying Local Weights to Factorized Part Encodings Li et al. [10] proposed sketch, a combination of random noise and features of the original data, produced by transforming vectors from the noise space to a basis matrix space in NMF. Following [10], we apply the pre-computed local weights to the part encodings that are factorized by the projection matrices (Fig. 1-(b)). Each pre-computed local weight is multiplied by each latent vector. Thanks to this operation, each factorized latent vector has a localized weight, and the encodings lie on a part-based subspace. We describe this process schematically in Fig. 3.

4 Implementation

To obtain large facial mesh data, we used the AFLW2000-3D dataset [11] containing 2,000 3D faces having 53,215 vertices each face. All faces are in full correspondence and generated by the Basel Face Model [12] without pose variations. The dataset was divided into a training set and a test set with 1,780

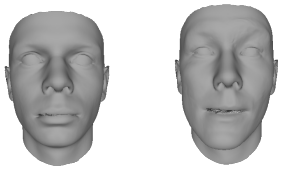
faces and 220 faces, respectively. Our proposed synthesis model has a similar architecture like CoMA [3]. With the basic autoencoder architecture, we add the projection matrix layer that we explained in Sec. 3.2. To optimize the networks, we exploit the L_1 and cycle loss [2]. We trained our model for 300 epochs with a batch size of 32. The dimension of the latent vector was 64. We followed Ranjan et al. [3] in terms of other hyper-parameters. We used PyTorch [13] and PyTorch Geometric [14] to implement our model and conducted all experiments with NVIDIA Titan RTX GPU 24GB.

5 Experimental Results

The experimental results of our proposed model are described in this section. We present the practicality of our model with generation tasks. In all experiments, we set the number of face parts, K, as 4. An extended version of this paper contains more implementation details and additional ablation test about the proposed model [4].

5.1 Generation Results

In this experiment, we tested the part manipulation results by applying interpolation between source and target as shown in Fig. 4. We interpolated the source's part encodings to the target's corresponding part encodings obtained by factorized latent vectors described in Sec. 3.2.



(a) Source (b) Target

Fig. 4: Source and target face

Part manipulation Fig. 5 shows that as the respective part of the face influence changes, the other parts of the face are not affected. Plus, we expected that each row's changing part matches each local weight in the same row. As a result, we observed that each variation area corresponds to each local weight in Fig. 5. Color gradients in the variation area included visualizing the Hausdorff distance between the first face ($\alpha = \frac{1}{9}$) and the last face ($\alpha = \frac{8}{9}$) in each row. Each of them displays a variation of each interpolation more clearly.

Diversity Visualization To demonstrate the variety of data, we measured the diversity of generated data from our model. Using the trained encoder, we encoded 220 random faces from our training set and test set, respectively. Since our proposed model allows part manipulation and modification, we synthesized 220 faces by combining five source faces and 11 target faces for four parts. The result was visualized by projecting selected data onto a 2D plane using PCA and t-SNE [15], shown in Fig. 6. We displayed all encoded faces as markers and summarized them with ellipses. Here, there are three types of encoding: training set (red), test set (yellow), and part synthesis (green).

In Fig. 6 - (a), we can discern that our synthesis sample area (green ellipse) involves both areas of the training set and test set (red and yellow ellipses) in the 2D PCA plane. Fig. 6 - (b) presents this result more distinctively as the synthesis samples are also located in a wider region as well as the region of

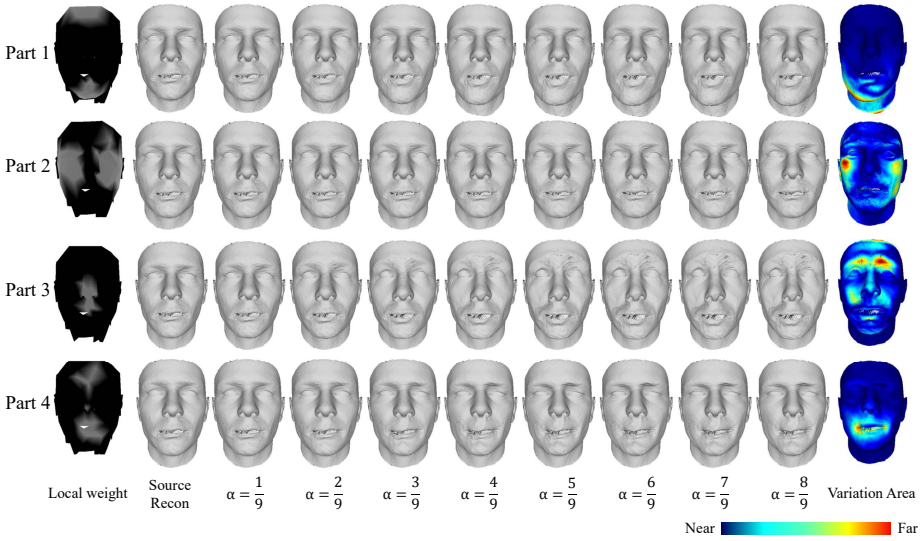
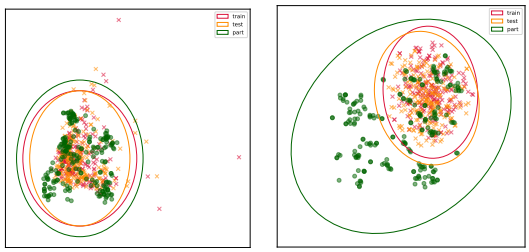


Fig. 5: Results of part interpolation

the training set and test set. In our visualizations, our synthesis samples (green) cover wider areas in the encoding space. As a result, our proposed method shows a prominent performance to extend the model's representation ability.



(a) PCA (b) t-SNE

Fig. 6: Diversity Visualization

5.2 Discussion

Although our proposed model performed notable part manipulation and synthesis using a holistic generative approach, there are a few points that need further discussion. First, the correlation between the changing area of faces and local weights should be better addressed. Most changing areas generally reflect corresponding local weights features, but some include another part or ignores them. One possible reason for this is that projection matrices would cover unassigned areas by transforming part encodings to local weight's space. The other is the

natural quality of the dataset having correlations between facial features. In second, we multiply the part encodings in latent space and local weights in NMF. This approach seemed to work in our setting because the projection matrices transform part encodings to local weights' space. We have shown experimentally that our process works, but a more rigorous mathematical proof is still needed.

6 Conclusion

We proposed a locally weighted 3D generative face model using spectral convolution networks for a 3D mesh. Our model show improved expressiveness by manipulating the local parts of a face without explicit mesh segmentation. In future work, we would like to extend our model to apply other generative models i.e., VAE or GANs, to improve output's quality. Generating face textures with geometry also would express the quality of outcomes better. Besides, it would be worthwhile to study part-based representation to improve the proposed local weights to develop the model's synthesis ability.

Acknowledgements This project was supported in part by the ITRC/IITP program (IITP-2021-0-01460) and the NRF (2017R1A2B3012701 and 2021R1A4A1032582) in South Korea. Y.-J. Kim is the corresponding author.

References

1. Nadav Schor, Oren Katzir, Hao Zhang, and Daniel Cohen-Or. Componet: Learning to generate the unseen by part synthesis and composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8759–8768, 2019.
2. Anastasia Dubrovina, Fei Xia, Panos Achlioptas, Mira Shalah, Raphaël Groscot, and Leonidas J Guibas. Composite shape modeling via latent space factorization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8140–8149, 2019.
3. Anurag Ranjan, Timo Bolkart, Soubhik Sanyal, and Michael J Black. Generating 3d faces using convolutional mesh autoencoders. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 704–720, 2018.
4. Minyoung Kim. Face geometry synthesis using locally weighted autoencoder. Master's thesis, Ewha Womans University, 2021.
5. Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.
6. J Rafael Tena, Fernando De la Torre, and Iain Matthews. Interactive region-based linear 3d face models. In *ACM SIGGRAPH 2011 papers*, pages 1–10. 2011.
7. Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv preprint arXiv:1606.09375*, 2016.
8. Vamsi K Potluru, Sergey M Plis, Jonathan Le Roux, Barak A Pearlmutter, Vince D Calhoun, and Thomas P Hayes. Block coordinate descent for sparse nmf. *arXiv preprint arXiv:1301.3527*, 2013.

9. Tim McGraw, Jisun Kang, and Donald Herring. Sparse non-negative matrix factorization for mesh segmentation. *International Journal of Image and Graphics*, 16(01):1650004, 2016.
10. Wei Li, Linchuan Xu, Zhixuan Liang, Senzhang Wang, Jiannong Cao, Chao Ma, and Xiaohui Cui. Sketch-then-edit generative adversarial network. *Knowledge-Based Systems*, 203:106102, 2020.
11. Xiangyu Zhu, Zhen Lei, Xiaoming Liu, Hailin Shi, and Stan Z Li. Face alignment across large poses: A 3d solution. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 146–155, 2016.
12. Pascal Paysan, Reinhard Knothe, Brian Amberg, Sami Romdhani, and Thomas Vetter. A 3d face model for pose and illumination invariant face recognition. In *2009 sixth IEEE international conference on advanced video and signal based surveillance*, pages 296–301. Ieee, 2009.
13. Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.
14. Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*, 2019.
15. Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.