Deep Facial Non-Rigid Multi-View Stereo
Supplementary Materials

Ziqian Bai[1] Zhaopeng Cui[2] Jamal Ahmed Rahim[1] Xiaoming Liu[3] Ping Tan[1]
[1] Simon Fraser University [2] ETH Zürich [3] Michigan State University
{ziqianb, jrahim, pingtan}@sfu.ca, zhpcui@gmail.com, liuxm@cse.msu.edu

In this supplementary material, we provide more information of the pipeline details (Section 1), additional experiments (Section 2), as well as additional qualitative results on in-the-wild images and videos (Section 3).

1. Pipeline Details

1.1. Feature Extraction

For computing the optimization objective of appearance-consistency (*i.e.* $\mathbf{E}_a$) and generating the adaptive basis (*i.e.* $\mathbf{B}_{adap}$), two sets of multi-level feature maps $\{\mathbf{F}_i\}_{i=1}^{M}$ and $\{\mathbf{F}'_i\}_{i=1}^{M}$ are extracted by two *Feature Pyramid Network*s (FPNs) [8]. As shown in Figure 1, each image is firstly fed into a backbone network, which is a modified version of *Dilated Residual Network* (DRN-38) [12], where we remove all dilations and add stride-2 down-sampling at the beginning of each resolution level. Then several convolutional blocks with up-sampling and skip connection are applied to generate the final feature maps. Note that our two FPNs share the same backbone.

1.2. Adaptive Face Model Generation

To better leverage the prior knowledge of facial expressions, we use the linear expression basis [6] built from FaceWarehouse [1] to initialize the first level adaptive basis at the beginning of training. More specifically, we initialize the kernel weights of the last convolution layer (without bias) in the first level basis network $\mathcal{F}^1_{basis}$ to zeros, and add the expression basis to our adaptive basis after texture mapping, as shown in the red box in Figure 2. Note that we allow this expression basis to be updated during training, which we find helpful to improve the fidelity of our results. We do not explicitly include the expression basis in our face model since we find it harming the convergence of training.

The number of parameters for each basis is set to $K_{bfm} = 80$ and $K_{adap} = 64$. We use four Residual Blocks [7] with dilation 1, 2, 4, and 8 in the Siamese branch. After the max pooling along view dimension, three convolutional blocks are applied to generate the UV texture representation of the adaptive basis.

1.3. Pose Initialization

At the very beginning (i.e., level 0), the initial head pose is regressed by a pre-trained neural network. In principal, any pose regression network can be used here. In our implementation, we use DRN38 [12] as a feature extractor, on top of which we build Siamese and view-shared branches to regress per-view poses similar to Tewari *et al.* [10]. The network is trained with the landmark loss computed by transforming and projecting the mean face from *Basel Face Model* (BFM) [9].

2. Additional Experiments

2.1. Dimensionality of the Adaptive Basis

We also investigate how the dimension of the adaptive basis affects the reconstruction quality. As shown in Table 1, different dimensions often lead to similar results while higher dimension produces slightly more robust results. Eventually, we empirically set it as 64 for a good trade off between the quality of results and memory usage.

Table 1: Geometric errors on BU3DFE [11] w.r.t. different dimensionalities of the adaptive basis.

# of Dimensions	16	32	64
Mean (mm)	1.11	1.12	1.11
STD (mm)	0.32	0.30	0.29

3. Additional Results

3.1. Additional Qualitative Evaluations on VoxCeleb2

In this section, we provide additional qualitative evaluations on VoxCeleb2 dataset [3], as shown in Figure 3 and Figure 4. In general, our results outperform previous single- and multi-view methods in terms of alignment to images (*e.g.*, 1^{st} row in Figure 4), and medium-level geometry details (*e.g.*, 1^{st} row in Figure 3).

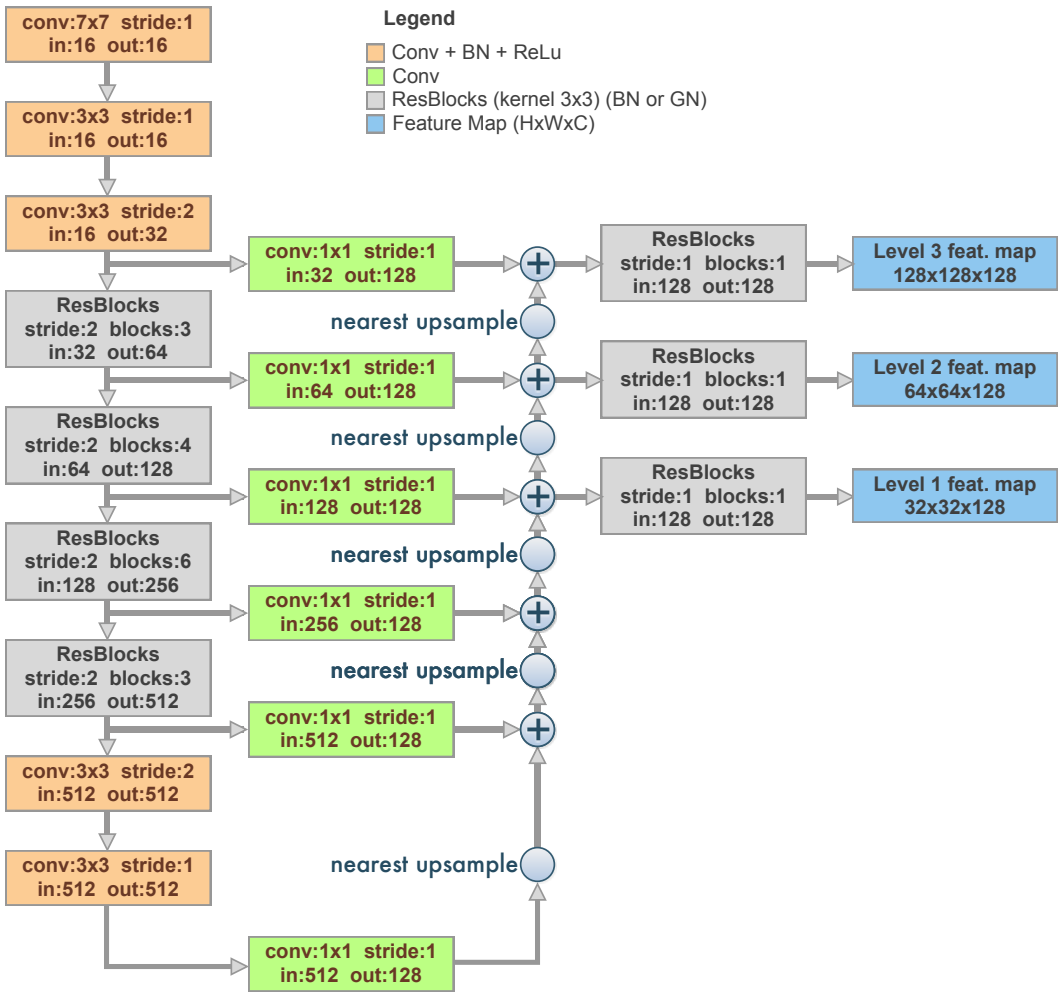

Figure 1: Feature Pyramid Networks Architecture.

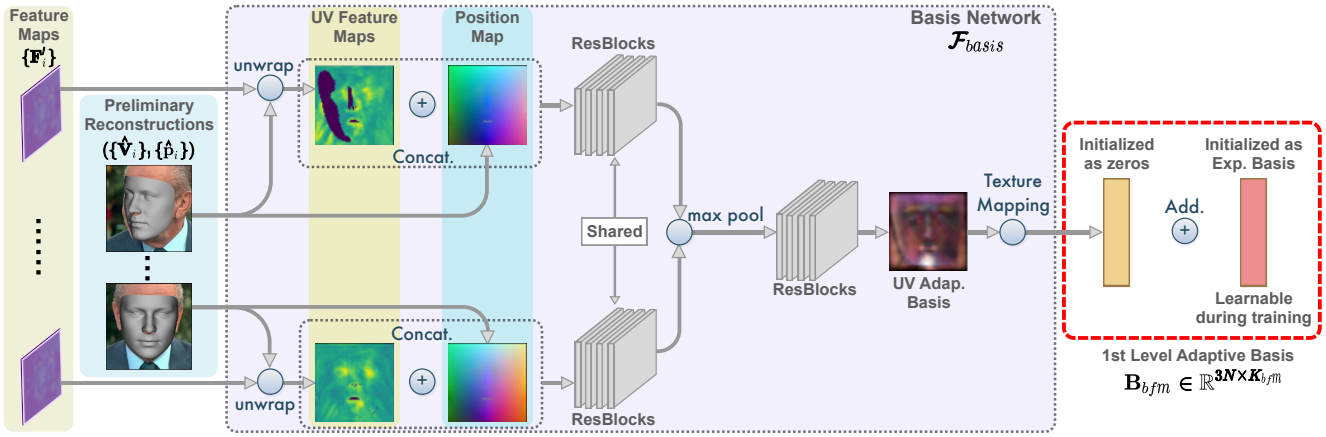

Figure 2: Detail pipeline of the 1st level Adaptive Basis. Details are highlighted with red box.

Note that (self-) occlusions may negatively affect our results, leading to misalignment (*e.g.*, 5^{th} row in Figure 4) and unfaithful geometry (*e.g.*, 8^{th} row in Figure 4). This can be alleviated by including more views and considering

face segmentation when computing the objectives for the optimization.

3.2. Video Results

We also test our method on three video sequences from VoxCeleb2 [3], including the comparison with Deng *et al*. [4]. We start with the initial 2 frames and cache them. Then, for each incoming frame, we first compute its head pose using the pose regression network, and choose the cached frame whose head pose has a larger difference in yaw angle with the incoming frame. We then use these 2 frames to perform the reconstruction. After that, if the yaw angle of head pose in the incoming frame is closer to $+40^\circ$ or -40°, we replace the corresponding cached frame with the incoming one. This head pose selection process is to ensure sufficient pose-coverage. From the video, we can see that our method could generate more temporally consistent and accurate 3D facial geometry compared to Deng *et al*. [4].

References

[1] Chen Cao, Yanlin Weng, Shun Zhou, Yiying Tong, and Kun Zhou. Facewarehouse: A 3d facial expression database for visual computing. *IEEE Trans. on Visualization and Computer Graphics (TVCG)*, 20(3):413–425, 2013. 1

[2] Anpei Chen, Zhang Chen, Guli Zhang, Kenny Mitchell, and Jingyi Yu. Photo-realistic facial details synthesis from single image. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 4, 5

[3] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. Voxceleb2: Deep speaker recognition. *Proc. Interspeech 2018*, pages 1086–1090, 2018. 1, 3

[4] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde Jia, and Xin Tong. Accurate 3d face reconstruction with weakly-supervised learning: From single image to image set. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 3

[5] Yao Feng, Fan Wu, Xiaohu Shao, Yanfeng Wang, and Xi Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *Proc. of European Conference on Computer Vision (ECCV)*, pages 534–551, 2018. 4, 5

[6] Yudong Guo, Jianfei Cai, Boyi Jiang, Jianmin Zheng, et al. Cnn-based real-time dense face reconstruction with inverse-rendered photo-realistic face images. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 41(6):1294–1307, 2018. 1

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1

[8] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 2117–2125, 2017. 1

[9] 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. 1

[10] Ayush Tewari, Florian Bernard, Pablo Garrido, Gaurav Bharaj, Mohamed Elgharib, Hans-Peter Seidel, Patrick Pérez, Michael Zollhofer, and Christian Theobalt. Fml: face model learning from videos. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, pages 10812–10822, 2019. 1, 4, 5

[11] Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang, and Matthew J Rosato. A 3d facial expression database for facial behavior research. In *7th international conference on automatic face and gesture recognition (FGR06)*, pages 211–216. IEEE, 2006. 1

[12] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *Proc. of Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

Figure 3: Qualitative comparison with Feng *et al.* [5], Chen *et al.* [2], and Tewari *et al.* [10]. For two-view methods, images of two consecutive rows are input together.

Figure 4: Qualitative comparison with Feng *et al.* [5], Chen *et al.* [2], and Tewari *et al.* [10]. For two-view methods, images of two consecutive rows are input together.