Gaussian Eigen Models for Human Heads – Supplemental Document –

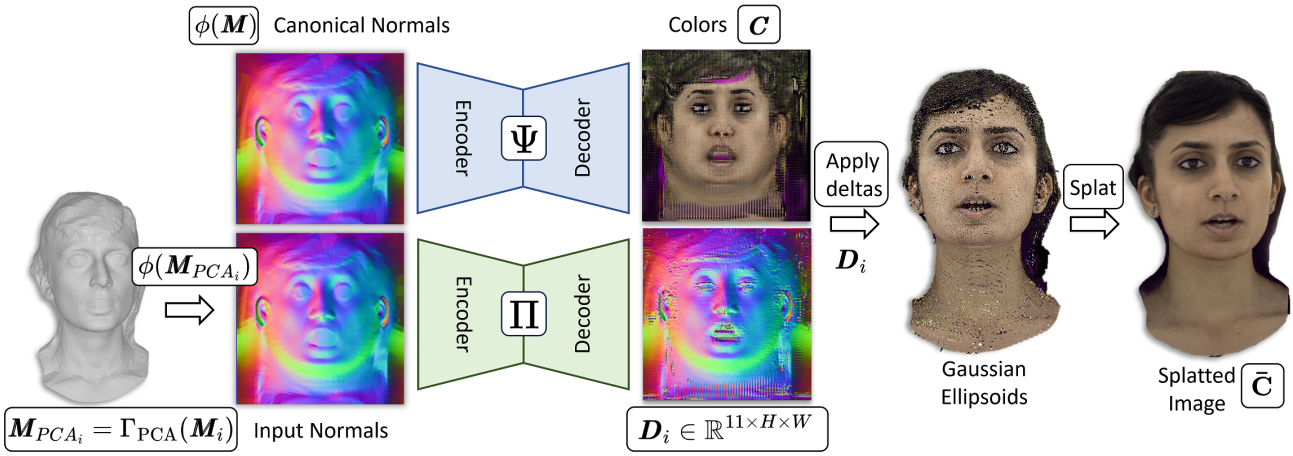


Figure 1. **CNN-based Gaussian Avatar Pipeline.** Our CNN model produces delta Gaussian maps $\boldsymbol{D}_i$ [20, 25] and static color $\boldsymbol{C}$ from a multi-view video. Similarly to Animatable Gaussians [20], we constrain the network to operate in a reduced linear space, i.e., the per-frame mesh $\boldsymbol{M}_i$ is projected on a PCA basis $\boldsymbol{M}_{PCA_i} = \Gamma_{\text{PCA}}(\boldsymbol{M}_i)$ which is then input to the network Π after converting the mesh to a normal map $\phi(\boldsymbol{M}_{PCA_i})$. The static color network is conditioned on neutral mesh $\boldsymbol{M}$.

A. 3D Gaussian Splatting Preliminaries

3D Gaussian Splatting (3DGS) [14] is an alternative approach to Neural Radiance Field (NeRF) [22] for static multi-view scene reconstruction and rendering under novel view. Kerbl et al. [14] parameterize the space as scaled 3D Gaussians [18, 37] with a 3D covariance matrix $\mathbf{\Sigma}$ and mean μ:

$$G(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\mu)^T\mathbf{\Sigma}^{-1}(\mathbf{x}-\mu)}. \tag{1}$$

To render this representation, Zwicker et al. [43] employ the projection of 3D Gaussians onto the image plane using the formula $\mathbf{\Sigma}' = \mathbf{AW\Sigma W}^T\mathbf{A}^T$, where $\mathbf{\Sigma}'$ represents the covariance matrix in 2D space. Here, $\mathbf{W}$ denotes the view transformation, and $\mathbf{A}$ represents the projective transformation. To avoid direct optimization of the covariance matrix $\mathbf{\Sigma}$ which must be positive semidefinite, Kerbl et al. [14] use scale $\mathbf{S}$ and rotation $\mathbf{R}$ which equivalently describes 3D Gaussian as a 3D ellipsoid $\mathbf{\Sigma} = \mathbf{RSS}^T\mathbf{R}^T$. Finally, 3DGS follows Ramamoorthi et al. [28] to approximate the diffuse part of the BRDF [8] as spherical harmonics (SH) to model global illumination and view-dependent color. Four bands of SH are used which results in a 48 elements vector.

B. Appearance Maps Generator

Similarly to StyleAvatar [36] and Animatable Gaussians [20], we use a StyleGAN-based [13] encoder and decoder network for this image translation. However, in contrast to Animatable Gaussians [20], we propose a more lightweight image translation pipeline, where we reduce the number of encoders from three to two and the number of decoders from six to two, and decrease the size of the StyleGAN [13] decoder. To efficiently use the 2D map space, we do not use projective textures from meshes [36], but a UV parametrization which reduces half of the decoders in comparison to StyleAvatar. Moreover, we do not use triplets of the encoder-decoder, as we combine all properties of the Gaussians into one map $\boldsymbol{G}_i$ following ASH [25]. Therefore, we define our network as follows:

$$\begin{aligned} \boldsymbol{D}_i &: \Pi(\phi(\Gamma_{\text{PCA}}(\boldsymbol{M}_i))), \\ \boldsymbol{C} &: \Psi(\phi(\boldsymbol{M})), \end{aligned} \tag{2}$$

where $\boldsymbol{D}_i \in \mathbb{R}^{11\times H\times W}$ is a map containing the delta for positions $\Delta_{pos} \in \mathbb{R}^3$, rotation $\Delta_{rot} \in \mathbb{R}^4$, scale $\Delta_{scale} \in \mathbb{R}^3$, and opacity $\Delta_{op} \in \mathbb{R}$. The colors $\boldsymbol{C} \in \mathbb{R}^{3\times H\times W}$ of the Gaussians are predicted using the normal map $\phi(\boldsymbol{M})$ of the canonical mesh $\boldsymbol{M}$ (ϕ is the normal map extractor from a mesh). Similar to Animatable Gaussians [20], we use a PCA layer Γ_{PCA} which serves as a low-pass regularization filter for the input. Γ_{PCA} is built by using PCA on the meshes $\boldsymbol{M}_i$ for the training frames and 16 principle components are used as the basis. For the training, we use the projection of each incoming mesh $\boldsymbol{M}_i$ on the PCA manifold $\boldsymbol{M}_{PCA_i} = \Gamma_{\text{PCA}}(\boldsymbol{M}_i)$. Optionally we use a feature texture that is concatenated with rasterized normals for the $\boldsymbol{D}_i$ conditioning.

The final Gaussian map $\boldsymbol{G}_i \in \mathbb{R}^{11\times H\times W}$ is obtained by applying the deltas to the canonical Gaussians $\boldsymbol{G}$ [20, 42]. The deformed position of 3D means is computed as

Ablation	L1 ↓	LIPIS ↓	PSNR ↑	SSIM ↑
Ours	0.0181	0.1171	24.2997	**0.9134**
Absolute	0.0181	0.1172	**24.3065**	0.9130
FLAME	0.0181	0.1169	24.2910	0.9130
EMOCA [4]	**0.0180**	**0.1165**	24.2715	0.9132
DECA [5]	0.0184	0.1189	24.2181	0.9127

Table 1. We performed an ablation study of our regressor using self-reenactment tasks. In this study, we tested configurations that used only EMOCA or DECA features, as well as a version where the EMOCA regressed FLAME expressions. Lastly, we show the effect of using absolute features instead of the relative ones used in Ours (features relative to the neutral face).

$\boldsymbol{T}_i(\boldsymbol{M} + \boldsymbol{D}_{i_{0:3}})$. One major distinction compared to AG [20] is the way how the transformation from the canonical space to the deformed space is handled. We employ deformation gradients, following Sumner et al. [32]. This approach allows for greater flexibility regarding input meshes, provided they maintain full correspondence.

Given a mesh $\boldsymbol{M}_{PCA_i}$ for the frame i, we define the deformation gradients as $\mathbf{J}_j = \hat{\mathbf{E}}_j \mathbf{E}_j^{-1}$, where $\hat{\mathbf{E}}_j \in \mathbb{R}^{3\times3}$ and $\mathbf{E}_j \in \mathbb{R}^{3\times3}$ contain the Frenet frame (tangent, bi-tangent, normal) of the triangle j defined in deformed and canonical spaces, respectively. Using these deformation gradients and the known correspondences between the Gaussian map and the meshes, we transform the Gaussians from the canonical space to the deformed space.

Note that our color map $\mathbf{C}$ is static and does not model view-dependent effects; this means that we force the network to recover globally consistent colors for each Gaussian similar to a texture in the classic 3DMM. Therefore, $\boldsymbol{G}_i$ must model the wrinkles and self-shadows.

Finally, we use Gaussian splatting [14] to render the regressed Gaussian maps. We define the predicted color of pixel (u, v) as:

$$\bar{\mathbf{C}}_{u,v} = \sum_{i\in\mathcal{N}} \mathbf{c_i}\alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{3}$$

where $\mathbf{c}_i$ is the Gaussian color predicted by Ψ, $\mathcal{N}$ is the number of texels and α_i is predicted opacity per Gaussian.

B.1. Image-based Coefficents Regressor

Table 1 shows an ablation study in the context of input to our MLP regressor which predicts GEM coefficients. Figure 2 provides an additional qualitative comparison. Using a pre-trained decoder such as EMOCA demonstrates strong potential for cross-reenactment by leveraging robust priors. Future work will explore further methods for image-based control of GEM, with a potential approach being the incorporation of additional modalities, such as sound, into the EMOCA-based regressor.

B.2. CNN Training Details

The training objective of the CNN-based appearance model is defined as $\mathcal{L} = \mathcal{L}_{Color} + \mathcal{L}_{Reg}$. $\mathcal{L}_{Color}$ is a weighted sum of three different photo-metric losses between the rendered image $\bar{\mathbf{C}}$ and the ground truth $\mathbf{C}$:

$$\begin{aligned} \mathcal{L}_{Color} &= (1-\omega)\mathcal{L}_1 + \omega\mathcal{L}_{\text{D-SSIM}} + \zeta\mathcal{L}_{\text{VGG}}, \\ \mathcal{L}_{Reg} &= \lambda \sum_{j=1}^{N} \left\| \Delta_{pos_j} \right\|^2 + \gamma \sum_{j=1}^{N} \left\| s_{scale_j} \right\|^2, \end{aligned} \tag{4}$$

where $\omega = 0.2$, $\zeta = 0.0075$ (after 150k iterations steps and zero otherwise), $\mathcal{L}_{\text{D-SSIM}}$ is a structural dissimilarity loss, and $\mathcal{L}_{\text{VGG}}$ is the perceptual VGG loss. $\mathcal{L}_{Reg}$ regularizes position offsets Δ_{pos_j} and scales s_{scale_j} to stay small w.r.t. the input mesh. We train our model for 10^6 steps using Adam [15] with a learning rate $5\text{e}{-4}$ and a batch size of one which takes around 10h on a Nvidia RTX4090.

Our method and all the baselines were trained using the same multiview input data sourced from the dataset provided by Qian [27], which includes multiview images from the NeRSamble dataset [17] as well as tracked meshes.

C. Compression Ablation Study

In the domain of 3D Morphable Models, principle component analysis (PCA) emerges as a cornerstone approach, instrumental in crafting the foundational framework for capturing face expressions and shapes with remarkable fidelity [2, 19]. This methodology has been adopted with notable success, not only in modeling facial features but also in extrapolating the nuances of human bodies [21, 24, 26], and even in depicting intricate hand modeling [29].

Expanding upon this foundation, GEM proposes a novel technique involving an ensemble of eigenbases of 3D Gaussian attributes for achieving a photorealistic human head appearance. This representation exhibits significant adaptability concerning both quality and size, leveraging a fundamental trait of linear basis that proves beneficial when applied to diverse devices with varying capabilities in digital human applications. Figures 8 and 9 illustrate the qualitative and quantitative results of GEM. Notably, even under substantial compression (utilizing only ten principal components), our approach consistently yields high-quality outcomes. More examples can be found in Figures 10, 11, 12 and 13.

D. Human Head Avatar Compression

Human avatar compression is an important topic, but it is still in its early stages and not well-explored. For neural-based representations, there are methods to compress networks, such as pruning [9], quantization [12], or knowledge distillation [10], as well as small and compact MobileNets

Figure 2. Our experiments show that using pre-trained regressor like EMOCA [4] and DECA [5] work well for driving our GEM model. In this context, DECA and EMOCA refer to using either of the regressed feature vectors. FLAME represents the expression vectors regressed by EMOCA. *Relative* and *absolute* denote whether the EMOCA + DECA features are used directly or as relative changes from a neutral face.

Figure 3. The quality comparison to Gaussian Avatars [27] shows better performance even though our model is similar in size to its Gaussian cloud and we do **not** need an additional FLAME model which weighs 90MB.

[11]. Interestingly, in the latter context, GEM can be considered as a single-layer MLP without any activation function. Unfortunately, these methods still require an expensive forward pass and may not be well-suited for all commodity devices.

#Comp	128^2	256^2	512^2
10	7	28	113
30	20	83	333
50	34	138	553
Ours Net	82	109	178
GA StyleUnet	487	529	636

Table 2. Memory consumption (in MB with float32) of GEM depends on the texture resolution and number of components. Our model shows much better granularity compared to the fixed size of neural networks and can be adjusted on the fly depending on the hardware.

#Comp	128^2	256^2	512^2
10	31.47	31.90	31.80
30	33.46	34.30	34.26
50	33.79	34.75	34.73

Table 3. PSNR color error in dB for one actor with a different number of principle components and Gaussian map resolutions. Despite heavy compression (10 principal components), the avatar is still of high quality. More details are in the supplementary material.

The recently introduced Gaussian Avatars by Qian [27] also represents a form of avatar compression, though not in the primitives' space but rather in the geometry space, with Gaussians attached and deformed by triangles from a linear face model. However, this form of appearance representation is insufficient for capturing details such as wrinkles, as it rigidly adheres to FLAME rigging in the geometry space. Therefore, we advocate for different compression techniques like GEM, which can leverage more powerful representations and distill them into expressive, high-quality linear models. We hope that this project will open doors to different methods for efficiently storing and representing avatars.

E. Additional Dataset Evaluation

In Figure 14, we provide further evaluation of our approach using the Multiface dataset [38]. This dataset encompasses short sequences of facial expressions, ranging from "relaxed mouth open" to "show all teeth" or "jaw open huge smile." The expressions vary widely in length and complexity, presenting a considerable challenge for analysis. It's important to note that this dataset does not provide a parametric 3DMM; instead, it offers meshes in full correspondence. However, as mentioned in the main text, our method remains adaptable in this context. By leveraging the deformation gradient [32] to transform points from canonical space into deformed space, and assuming consistent UV parametrization of input meshes, we can successfully navigate between these spaces. As depicted in Figure 14, our network demonstrates the ability to extrapolate to novel expressions, even amidst highly challenging facial poses.

F. Broader Impact

Our project focuses on reconstructing a highly detailed human face avatar from multiview videos, enabling the extrapolation of expressions not originally captured. While our technology serves primarily constructive purposes, such as enriching telepresence or mixed reality applications, we acknowledge the potential risks of misuse. Therefore, we advocate for advancements in digital media forensics [30, 31] to aid in detecting synthetic media. Additionally, we emphasize the importance of conducting research in this area with transparency and openness, including the thorough disclosure of algorithmic methodologies, data origins, and models intended for research purposes.

G. Future Applications & Discussion

An interesting application venue for GEM would be a combination of audio-driven methods with the appearance offered by our methods. Ng et at. [23] presented photorealistic audio-driven full-body avatars. Despite impressive results, the face region still does not fully convey expressions and lacks realism. One way of improving it would be incorporating recent progress in audio-driven geometry [1, 3, 34, 35] with a dedicated appearance model offered by GEM and our image-space regressor Figure 6.

Moreover, our neural network based appearance model uses meshes to obtain normal maps as input to the Gaussian map regressor (similar to the baselines). However, meshes are limited by resolution and expressiveness, one way of improving on that would be to use NPHM by Giebenhain et al. [6] and the follow-up work [7, 16, 33] to further increase the expressiveness of the model by explicitly capturing regions like hair or teeth.

Figure 4. GEM applied on different avatar methods (AG and GA) and optimized using analysis-by-synthesis. Our method is universal and can be successfully used on point clouds and textures to distill a lightweight avatar.

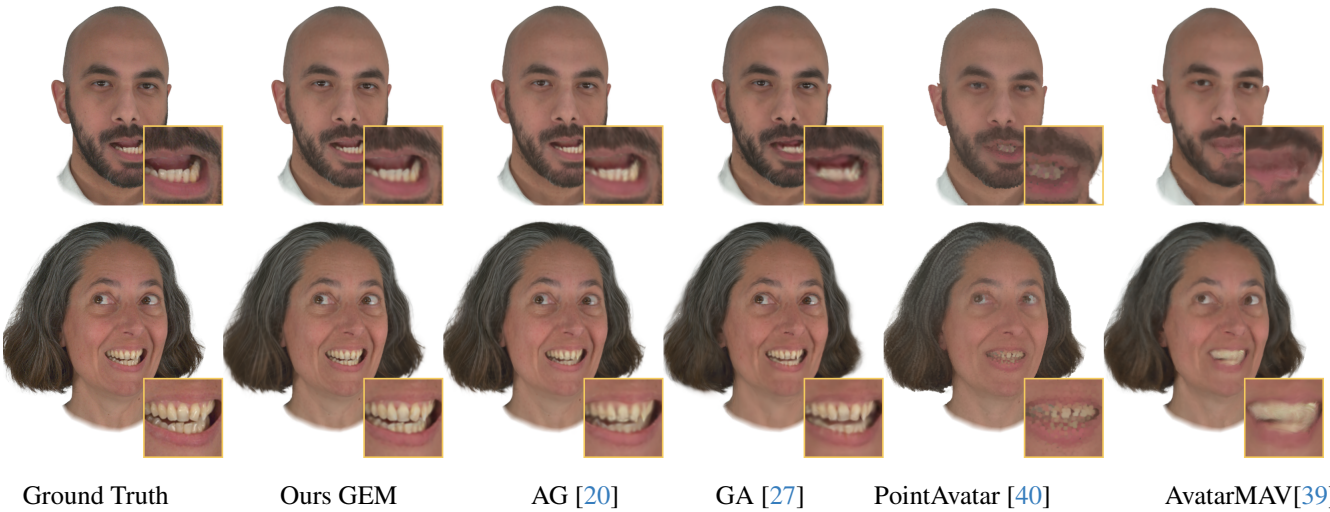


Figure 5. Additional baselines PointAvatar (PSNR: 25.8, SSIM: 0.893 LPIPS: 0.097) and AvatarMAV (PSNR: 29.5, SSIM: 0.913, LPIPS: 0.152) evaluated on the novel-view sequences.

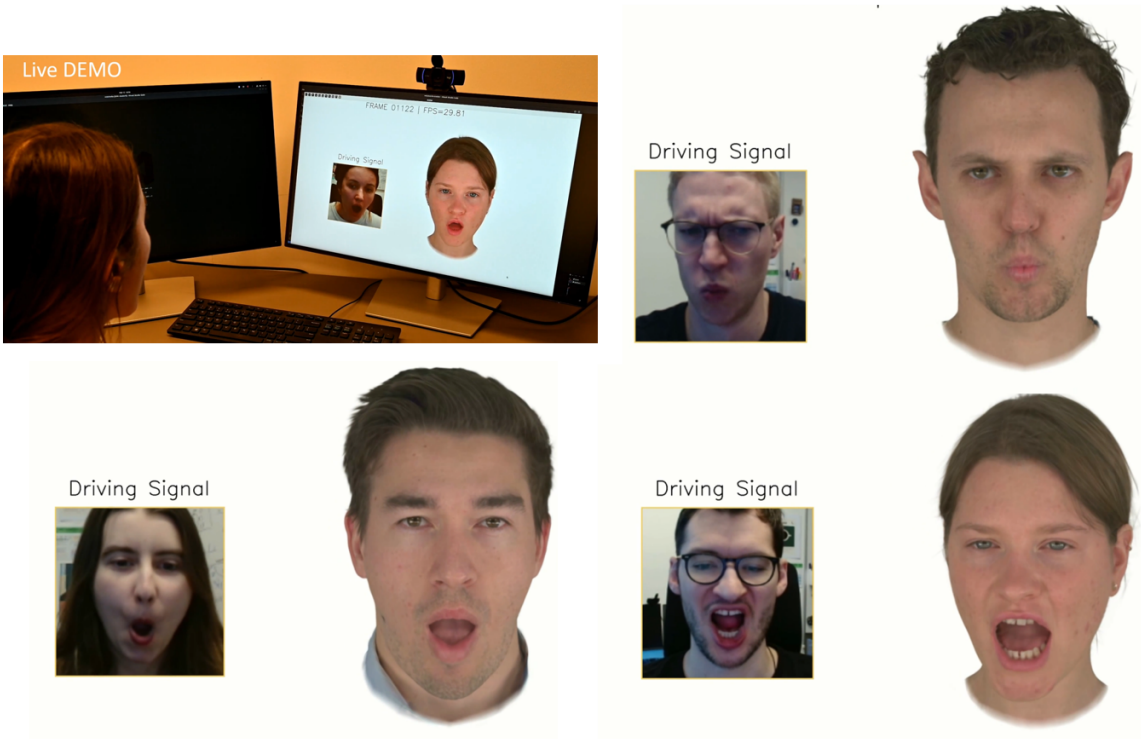
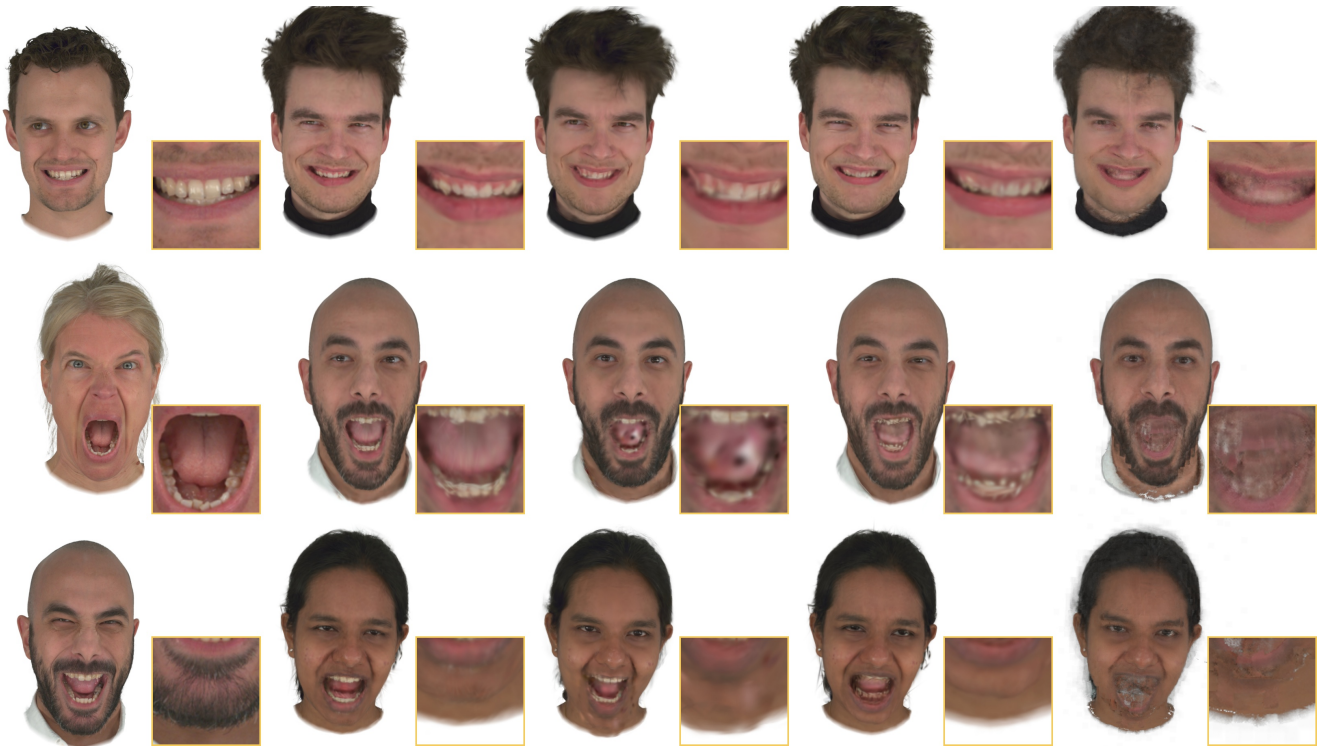


Figure 6. GEM can be effectively controlled in real-time by an image-space regressor which produces coefficients projected on the linear basis of a personalized GEM avatar.



Ground Truth Ours Net GA. [27] AG. [20] INSTA [41]

Figure 7. **Facial cross-person reenactment.** The person's expressions on the left are transferred to the respective avatars on the right. In this experiment, we are using relative expressions based on ground truth meshes from the dataset (FLAME-based meshes reconstructed from multi-view data). Note that this experiment does not apply to our GEM, since it is mesh-free.

Figure 8. Qualitative compression quality depending on the number of basis (10, 30, 50) and resolution of the Gaussian map (128^2, 256^2, 512^2).

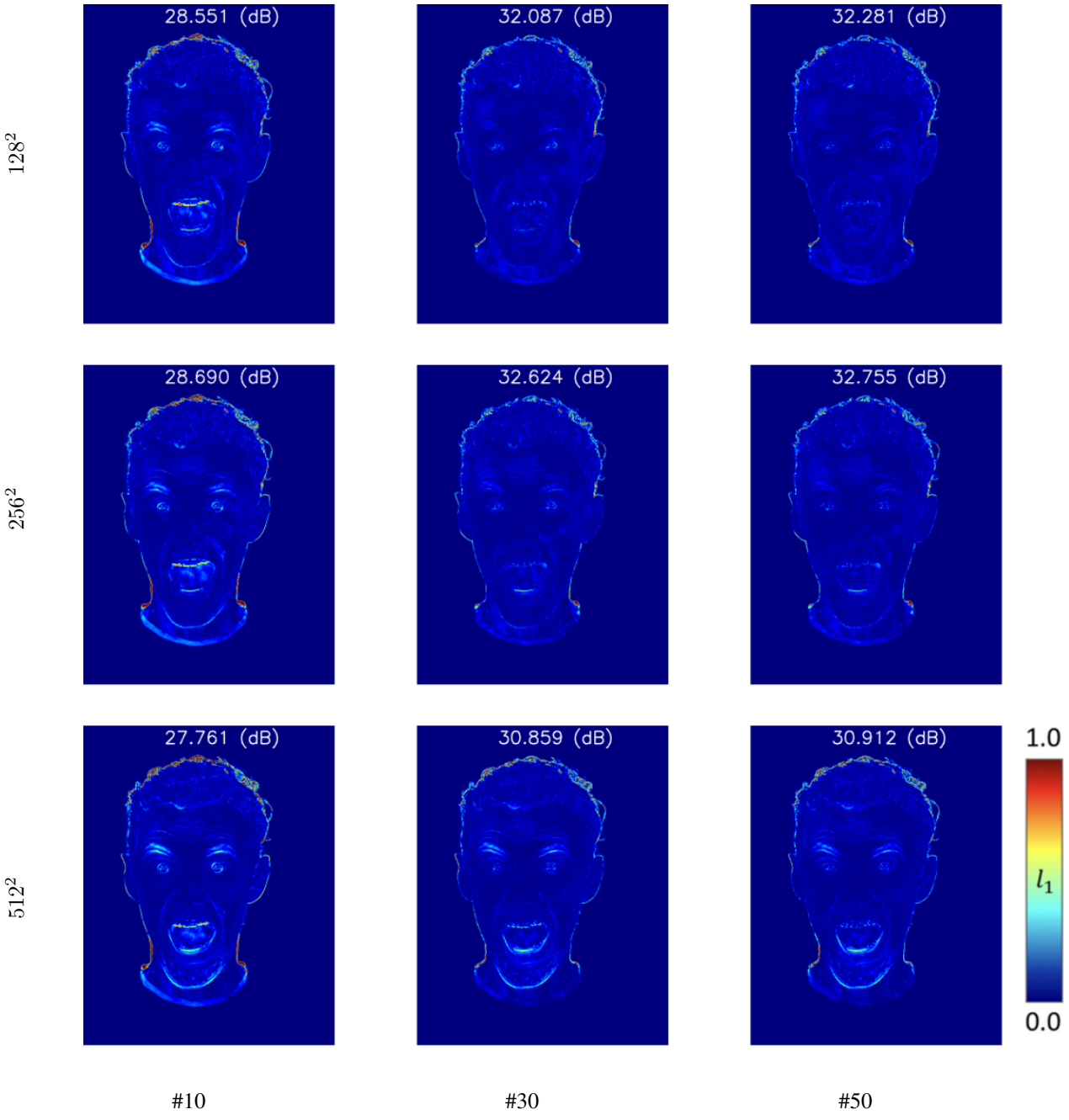


Figure 9. Compression error in PSNR (db) depending on the number of basis (10, 30, 50) and resolution of the Gaussian maps (128^2, 256^2, 512^2).

Figure 10. Qualitative compression quality depending on the number of basis (10, 30, 50) and resolution of the Gaussian map (128^2, 256^2, 512^2).

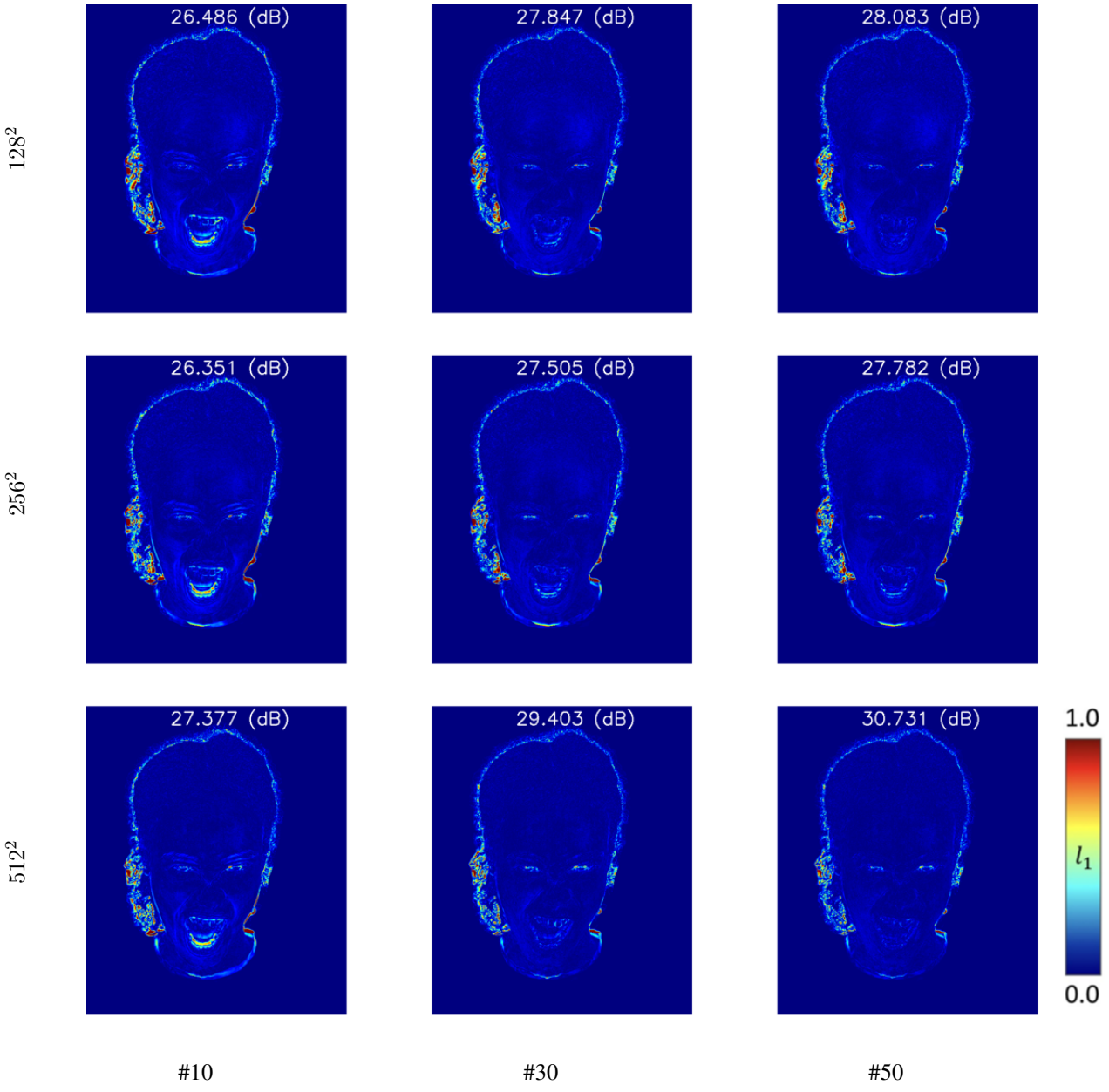


Figure 11. Compression error in PSNR (db) depending on the number of basis (10, 30, 50) and resolution of the Gaussian maps (128^2, 256^2, 512^2).

Figure 12. Qualitative compression quality depending on the number of basis (10, 30, 50) and resolution of the Gaussian map (128^2, 256^2, 512^2).

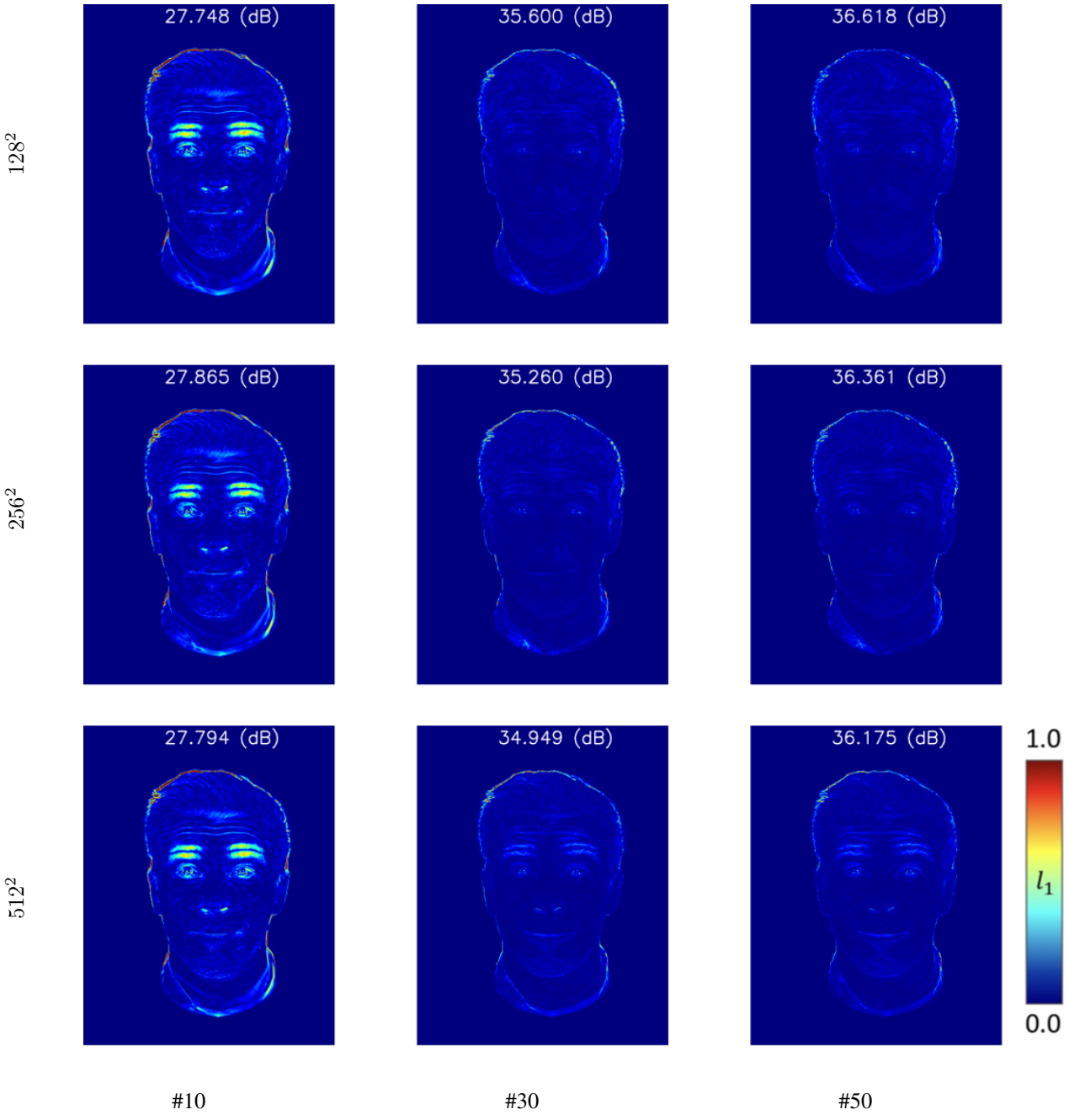
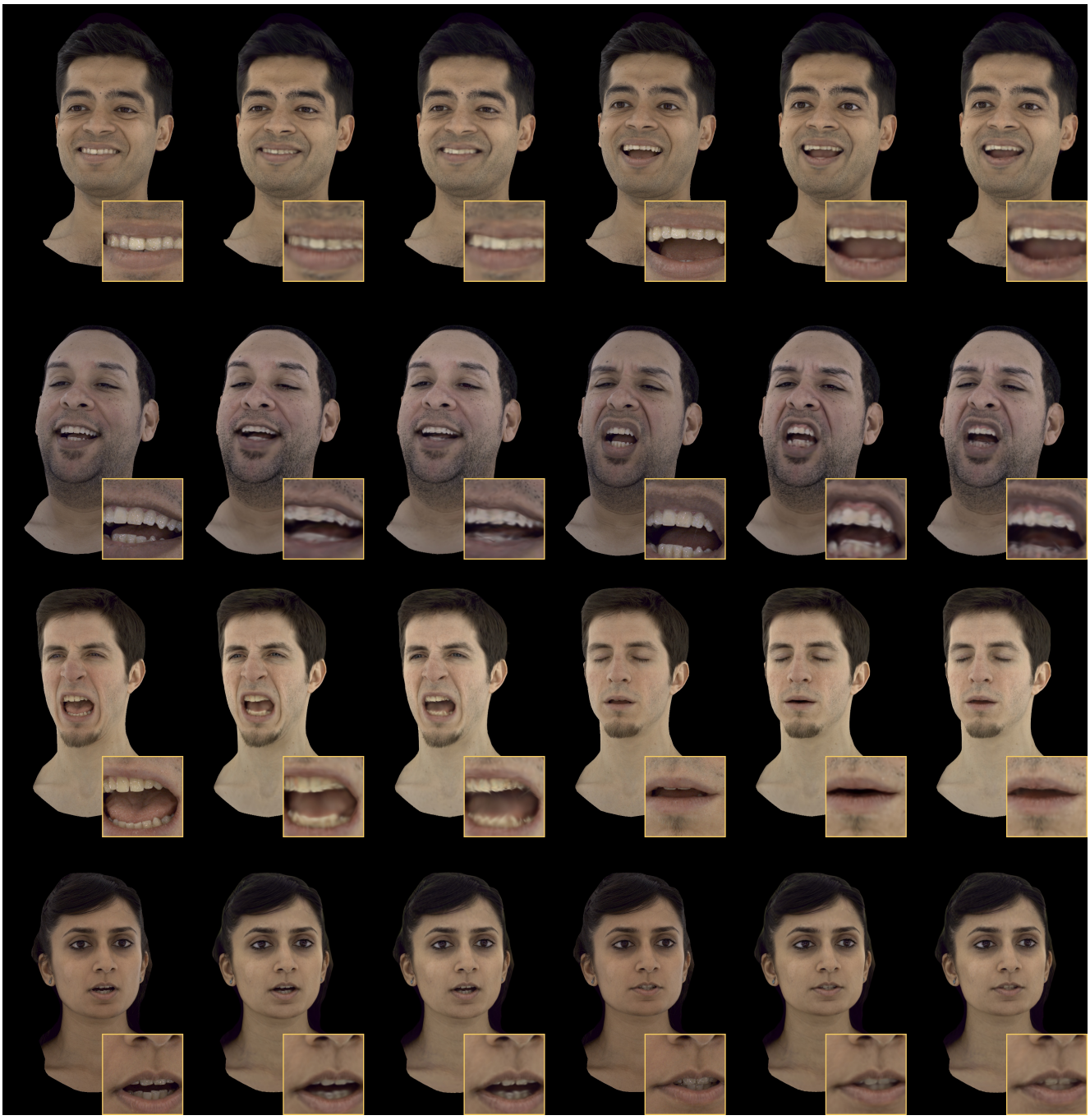


Figure 13. Compression error in PSNR (db) depending on the number of bases (10, 30, 50) and resolution of the Gaussian maps (128^2, 256^2, 512^2).



Ground Truth Ours Net Ours GEM Ground Truth Ours Net Ours GEM

Figure 14. The Multiface dataset, introduced by Wu et al. [38], comprises actors performing scripted expressions in short segments. A notable challenge arises due to the occurrence of several expressions, like "show all teeth," appearing only once in the dataset. This poses a difficulty during testing, particularly when the network is required to extrapolate. Here we showcase the outcomes of the test sequences to illustrate the effectiveness of our CNN-based network in capturing diverse and challenging facial poses, demonstrating its robustness despite the inherent complexity of the dataset.

References

[1] Shivangi Aneja, Justus Thies, Angela Dai, and Matthias Nießner. Facetalk: Audio-driven motion diffusion for neural parametric head models. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 4

[2] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *SIGGRAPH*, pages 187–194, 1999. 2

[3] Daniel Cudeiro, Timo Bolkart, Cassidy Laidlaw, Anurag Ranjan, and Michael Black. Capture, learning, and synthesis of 3D speaking styles. pages 10101–10111, 2019. 4

[4] Radek Danecek, Michael J. Black, and Timo Bolkart. EMOCA: Emotion driven monocular face capture and animation. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20279–20290, 2022. 2, 3

[5] Yao Feng, Haiwen Feng, Michael J. Black, and Timo Bolkart. Learning an animatable detailed 3D face model from in-the-wild images. *Transactions on Graphics (TOG)*, 40:1 – 13, 2020. 2, 3

[6] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Learning neural parametric head models. 2023. 4

[7] Simon Giebenhain, Tobias Kirschstein, Markos Georgopoulos, Martin Rünz, Lourdes Agapito, and Matthias Nießner. Mononphm: Dynamic head reconstruction from monoculuar videos. 2024. 4

[8] Cindy M. Goral, Kenneth E. Torrance, Donald P. Greenberg, and Bennett Battaile. Modeling the interaction of light between diffuse surfaces. *SIGGRAPH*, 1984. 1

[9] Song Han, Jeff Pool, John Tran, and William J. Dally. Learning both weights and connections for efficient neural networks, 2015. 2

[10] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015. 2

[11] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications, 2017. 4

[12] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference, 2017. 2

[13] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8107–8116, 2019. 1

[14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *Transactions on Graphics (TOG)*, 42:1 – 14, 2023. 1, 2

[15] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 2

[16] Tobias Kirschstein, Simon Giebenhain, and Matthias Nießner. Diffusionavatars: Deferred diffusion for high-fidelity 3d head avatars. *arXiv preprint arXiv:2311.18635*, 2023. 4

[17] Tobias Kirschstein, Shenhan Qian, Simon Giebenhain, Tim Walter, and Matthias Nießner. NeRSemble: Multi-view radiance field reconstruction of human heads. *Transactions on Graphics (TOG)*, 42:1 – 14, 2023. 2

[18] Georgios Kopanas, Julien Philip, Thomas Leimkühler, and George Drettakis. Point-based neural rendering with per-view optimization. *Computer Graphics Forum (CGF)*, 40, 2021. 1

[19] Tianye Li, Timo Bolkart, Michael. J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6):194:1–194:17, 2017. 2, 3

[20] Zhe Li, Zerong Zheng, Lizhen Wang, and Yebin Liu. Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 1, 2, 5, 6

[21] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. 34(6):248:1–248:16, 2015. 2

[22] 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 *European Conference on Computer Vision (ECCV)*, pages 405–421, 2020. 1

[23] Evonne Ng, Javier Romero, Timur Bagautdinov, Shaojie Bai, Trevor Darrell, Angjoo Kanazawa, and Alexander Richard. From audio to photoreal embodiment: Synthesizing humans in conversations. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 4

[24] Ahmed A A Osman, Timo Bolkart, Dimitrios Tzionas, and Michael J. Black. SUPR: A sparse unified part-based human body model. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[25] Haokai Pang, Heming Zhu, Adam Kortylewski, Christian Theobalt, and Marc Habermann. ASH: animatable gaussian splats for efficient and photoreal human rendering. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1165–1175. IEEE, 2024. 1

[26] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019. 2

[27] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. GaussianAvatars: Photorealistic head avatars with rigged 3D gaussians. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20299–20309, 2024. 2, 3, 4, 5, 6

[28] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. *SIGGRAPH*, 2001. 1

[29] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. 36(6), 2017. 2

[30] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics: A large-scale video dataset for forgery detection in human faces. *ArXiv*, 2018. 4

[31] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. Faceforensics++: Learning to detect manipulated facial images. *International Conference on Computer Vision (ICCV)*, pages 1–11, 2019. 4

[32] Robert W. Sumner and Jovan Popović. Deformation transfer for triangle meshes. *SIGGRAPH*, 2004. 2, 4

[33] Jiapeng Tang, Angela Dai, Yinyu Nie, Lev Markhasin, Justus Thies, and Matthias Niessner. Dphms: Diffusion parametric head models for depth-based tracking. 2024. 4

[34] Balamurugan Thambiraja, Sadegh Aliakbarian, Darren Cosker, and Justus Thies. 3DiFACE: Diffusion-based speech-driven 3D facial animation and editing. *ArXiv*, abs/2312.00870, 2023. 4

[35] Balamurugan Thambiraja, Ikhsanul Habibie, Sadegh Aliakbarian, Darren Cosker, Christian Theobalt, and Justus Thies. Imitator: Personalized speech-driven 3D facial animation. pages 20564–20574, 2023. 4

[36] Lizhen Wang, Xiaochen Zhao, Jingxiang Sun, Yuxiang Zhang, Hongwen Zhang, Tao Yu, and Yebin Liu. StyleAvatar: Real-time photo-realistic portrait avatar from a single video. In *SIGGRAPH Conference Papers (SA)*, pages 67:1–67:10, 2023. 1

[37] Yifan Wang, Felice Serena, Shihao Wu, Cengiz Öztireli, and Olga Sorkine-Hornung. Differentiable surface splatting for point-based geometry processing. *Transactions on Graphics (TOG)*, 38:1 – 14, 2019. 1

[38] Cheng-hsin Wuu, Ningyuan Zheng, Scott Ardisson, Rohan Bali, Danielle Belko, Eric Brockmeyer, Lucas Evans, Timothy Godisart, Hyowon Ha, Xuhua Huang, Alexander Hypes, Taylor Koska, Steven Krenn, Stephen Lombardi, Xiaomin Luo, Kevyn McPhail, Laura Millerschoen, Michal Perdoch, Mark Pitts, Alexander Richard, Jason Saragih, Junko Saragih, Takaaki Shiratori, Tomas Simon, Matt Stewart, Autumn Trimble, Xinshuo Weng, David Whitewolf, Chenglei Wu, Shoou-I Yu, and Yaser Sheikh. Multiface: A dataset for neural face rendering. In *arXiv*, 2022. 4, 13

[39] Yuelang Xu, Lizhen Wang, Xiaochen Zhao, Hongwen Zhang, and Yebin Liu. Avatarmav: Fast 3d head avatar reconstruction using motion-aware neural voxels. In *ACM SIGGRAPH 2023 Conference Proceedings*, 2023. 5

[40] Yufeng Zheng, Yifan Wang, Gordon Wetzstein, Michael J. Black, and Otmar Hilliges. PointAvatar: Deformable point-based head avatars from videos. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21057–21067, 2022. 5

[41] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Instant volumetric head avatars. *Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4574–4584, 2022. 6

[42] Wojciech Zielonka, Timur Bagautdinov, Shunsuke Saito, Michael Zollhöfer, Justus Thies, and Javier Romero. Drivable 3D gaussian avatars. In *International Conference on 3D Vision (3DV)*, 2025. 1

[43] Matthias Zwicker, Hans Rüdiger Pfister, Jeroen van Baar, and Markus H. Gross. Surface splatting. *SIGGRAPH*, pages 371–378, 2001. 1