AnimateMe: 4D Facial Expressions via Diffusion Models (Supplementary Material)

Dimitrios Gerogiannis, Foivos Paraperas Papantoniou, Rolandos Alexandros Potamias, Alexandros Lattas, Stylianos Moschoglou, Stylianos Ploumpis, and Stefanos Zafeiriou

Imperial College London, UK
{d.gerogiannis22, f.paraperas, r.potamias, a.lattas, s.moschoglou, s.ploumpis, s.zafeiriou}@imperial.ac.uk

In this document, we provide additional details about our method, that could not fit on the main paper due to page limitations, as well as a thorough ethical and limitations discussion. Moreover, we attach a video that briefly explains the proposed method, and shows additional results in video format, that highlight the effectiveness and quality of our method.

1 Additional Implementation Details

1.1 Preprocessing Details

To achieve a standardized animation length of 40 frames, despite our method's flexibility in generating varying frame counts, we employ a combination of selection and interpolation techniques. For sequences exceeding 40 frames, we select the most distinct frames in terms of their differences from consecutive frames. On the other hand, for sequences with fewer than 40 frames, we employ interpolation between the frames that exhibit the greatest differences, thereby seamlessly expanding to the required frame count.

Regarding the extremeness factors, they are derived by calculating deformations at key facial landmarks from neutral to apex frames for each animation. These factors are then normalized against the maximum extremeness factor within the same expression category, as mentioned in the main paper.

1.2 Dataset Splitting Selection

Regarding dataset splitting, our approach diverges from that of [1]. Whereas [1] utilizes the first subsequence from each subject and expression from their divided subsequences for testing and allocate the rest for training, we adopt the split methodology from the subject-independent CoMA [4] experiments. This involves training on nine identities and testing on three, encompassing all their expression animations. This decision was made to ensure the model remains unexposed to the specific landmarks or deformation patterns of the test identities during training, thus avoiding potential overfitting. Ideally, the models should be completely unfamiliar with the test data to ensure fair evaluation conditions. Therefore, our choice aligns with the goals of evaluating model generalizability, offering a more accurate measure of its performance on unseen data.

2 Consistent Noise Sampling

2.1 Algorithm

For the sake of completeness, we also provide the analytical algorithm for consistent noise sampling, ensuring that the notation remains consistent with that used in the main paper.

Algorithm 1 Consistent Noise Sampling

Get the neutral mesh to be animated: $\mathbf{x_0} \in \mathbb{R}^{N\times 3}$
Get the expression progression signal: $\mathbf{E} = \{\mathbf{e_i} : i = 0, \ldots, K-1\}$

$\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
$\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), t = T, \ldots, 2$
$\mathbf{z}_1 = \mathbf{0}$

Sample the expression progression for the first frame: $\mathbf{e_0}$
$\hat{\mathbf{d}}_\mathbf{0}^{\,\mathbf{T}} = \boldsymbol{\epsilon}$
for $t = T$ to $t_s + 1$ **do**
    $\hat{\mathbf{d}}_\mathbf{0}^{\,\mathbf{t-1}} = \frac{1}{\sqrt{a_t}} \left(\hat{\mathbf{d}}_\mathbf{0}^{\,\mathbf{t}} - \frac{1-a_t}{\sqrt{1-\bar{a}_t}} \mathbf{s}_\theta(\hat{\mathbf{d}}_\mathbf{0}^{\,\mathbf{t}}, t, \mathbf{e_0}) \right) + \sigma_t \mathbf{z_t}$

$\hat{\mathbf{d}}_\mathbf{s} = \hat{\mathbf{d}}_\mathbf{0}^{\,\mathbf{t_s}}$
for $k = 0$ to $K-1$ **in parallel do**
    Sample the expression progression for the current frame: $\mathbf{e_i}$
    $\hat{\mathbf{d}}_\mathbf{k}^{\,\mathbf{t_s}} = \hat{\mathbf{d}}_\mathbf{s}$
    for $t = t_s$ to 1 **do**
        $\hat{\mathbf{d}}_\mathbf{k}^{\,\mathbf{t-1}} = \frac{1}{\sqrt{a_t}} \left(\hat{\mathbf{d}}_\mathbf{k}^{\,\mathbf{t}} - \frac{1-a_t}{\sqrt{1-\bar{a}_t}} \mathbf{s}_\theta(\hat{\mathbf{d}}_\mathbf{k}^{\,\mathbf{t}}, t, \mathbf{e_k}) \right) + \sigma_t \mathbf{z_t}$
    Apply the deformations to the neutral mesh: $\hat{\mathbf{x_k}} = \mathbf{x_0} + \hat{\mathbf{d}}_\mathbf{k}^{\,\mathbf{0}}$

2.2 Ablation Study

During experimentation, different t_s values were explored, settling on $t_s = 400$ as a trade-off between sufficient denoising steps for generating frames and faster inference. While additional denoising steps enhance animation quality, extending beyond 400 steps yielded no significant improvement. This approach halved the inference time from 26.86 to 12.83 seconds. While slower, it's comparable to GAN-based methods, illustrating the well-known trade-off between diversity and quality versus inference time in diffusion models.

3 Architectural Details

Besides transitioning from an MLP to an SCN for the denoising model architecture, additional adjustments have been implemented to optimize mesh

generation. The model is enhanced through the integration of learnable index embeddings with a small dimensionality, which are added to the point clouds as additional features for each point. These embeddings not only improve the smoothness of the generated point clouds but also assist with the maintenance of the mesh's connectivity. Further refinement is achieved with the use of learnable timestep embeddings of increased dimensions. This adaptation allows for the capture of complex patterns essential for the success of the diffusion process. Additionally, the architecture is designed to prioritize the preservation of details and the recognition of complex patterns in 3D data. It features an increase in both the number of filters and the sequence length progressively deeper into the network. This strategy ensures a hierarchical representation of the data, enabling the capture of increasingly abstract and complex features. Finally, to maintain the complexity and detail of the 3D mesh data across all layers, the architecture avoids downsampling. This decision ensures that no vital information is lost throughout the denoising process.

4 Additional Quantitative Results

We also provide quantitative results for a setup we term the "expression split" in Tab. 1. In this setup, we train the model on all subjects but exclude certain expressions for some of these subjects, which are then used for testing. Essentially, the model sees all identities during training but not all expressions for some individuals. This setup tries to mimic an expression-independent split, commonly used in reconstruction tasks. However, the expression-independent split is not applicable for our task, since it is a generation task.

Table 1: Classification accuracy and specificity error for expression split.

Method	Accuracy (%)	Specificity (mm)
GT	75.13	0
MO3DGAN [1]	65.03	2.21
LSTM [3]	46.14	3.18
Ours-Extreme	73.62	1.83
Ours-Local	70.67	1.78
Ours-Varying	76.25	1.66

5 Additional Qualitative Results

We provide additional qualitative results to showcase the diversity of the generated expressions of our model in Fig. 1. Please note that we only provide the last frame of the expression animations.

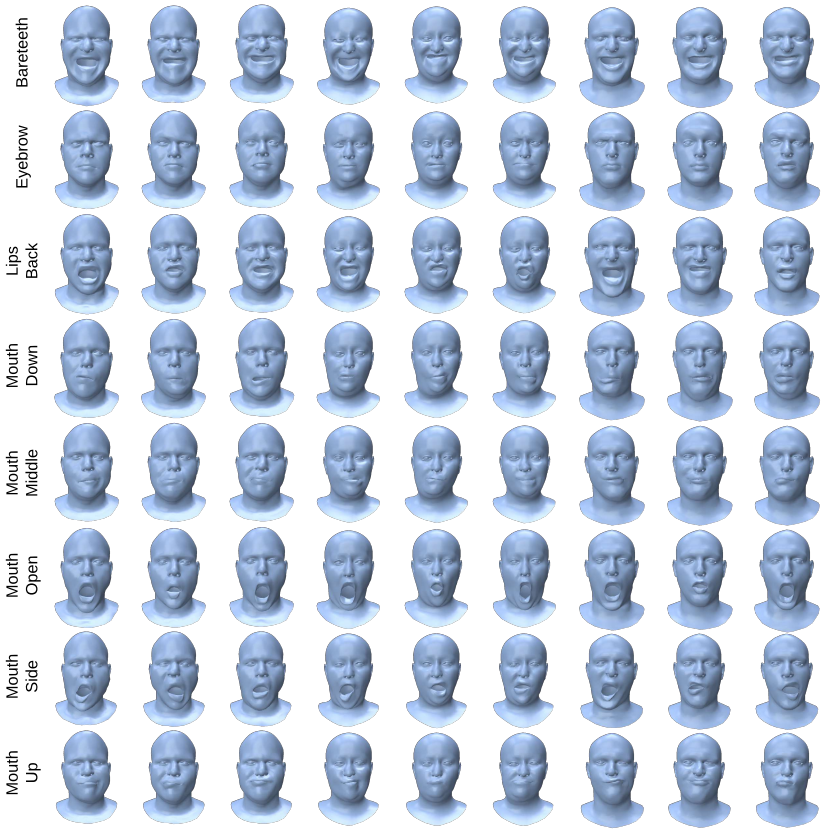

Fig. 1: Diversity of generated expressions.

6 User Study

To further validate our claims of superiority, we conducted a user study involving 15 participants. They were tasked with comparing videos featuring expression animations generated by both our proposed model and the MO3DGAN model, focusing on aspects of realism and animation smoothness. The results, as shown in Tab. 2, demonstrate that our method significantly outperforms MO3DGAN in both realism and animation smoothness.

Table 2: User study results on realism and animation smoothness.

	Realism		Animation Smoothness	
	Ours	MO3DGAN	**Ours**	MO3DGAN
Preference	**11**	4	**9**	6

7 Expression Interpolation

Our method facilitates expression interpolation by linearly interpolating between expression labels. This process allows for the generation of animations that transition smoothly between two expressions. The effectiveness of this approach is illustrated in Fig. 2, which demonstrates the intermediate states during the interpolation process.

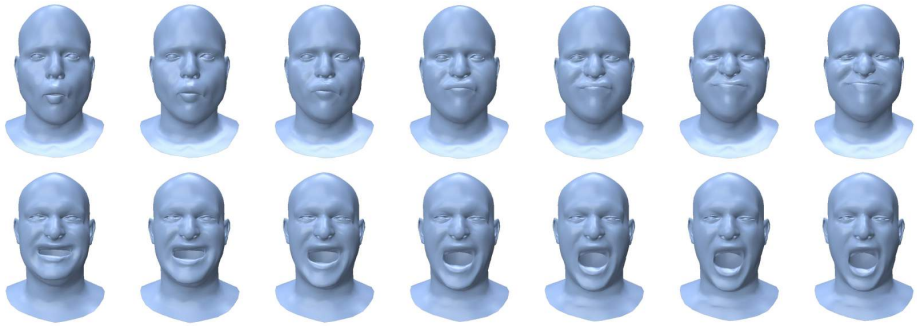

Fig. 2: Expression interpolation with intermediate states between two expression labels of specific intensities.

8 Method Limitations

We acknowledge that while our method represents a significant advancement in the generation of 4D facial expressions, surpassing previous works, it is not without its limitations. Firstly, our approach is inherently conditioned on an expression progression signal. This conditioning, despite enabling customizable generation, is not ideal due to its restrictive nature. Future iterations of our work could benefit from exploring additional conditioning options to enhance versatility. Secondly, our method's reliance on mesh representations, stemming from the use of a Graph Neural Network (GNN) based denoising model, limits its applicability to other 3D representation forms. This specialization, while effective, narrows the scope of our method's utility. Thirdly, the adoption of a diffusion approach inherently slows down our method, a drawback that becomes more pronounced with larger meshes. Nevertheless, in our textured experiment, downsampling high-resolution meshes (nearly 28K vertices) prior to processing and then upsampling the results has proven effective in mitigating speed issues. Future work could explore integrating DDIM [6] sampling with our diffusion model to accelerate generation. Additionally, while not a primary concern, the generation speed of our texture LDM [5] also presents challenges. However, by employing DDIM sampling with just 200 timesteps, we achieved very good results. Looking ahead, conditioning the LDM on the geometry of each frame

could significantly enhance the coherence between generated textures and geometries, marking a direction for future research. Finally, in its current version, our method generates expression animations starting from a neutral state, which is a limitation compared to [1,2]. Future extensions of our method could address this limitation by allowing animations to begin from an expressive state.

9 Ethical Limitations

Our method animates static facial meshes to match specific expressions, opening up new possibilities in digital media. However, it also presents ethical challenges, particularly concerning misuse and consent. The ease with which our method can animate faces raises concerns about its potential use in creating deepfakes. These manipulated 3D animations can mislead people or harm someone's reputation without their permission. Equally important is the issue of consent and ownership. Using someone's likeness to animate expressions without their clear approval crosses ethical boundaries, especially if those animations are used in ways the person wouldn't agree with. These ethical considerations highlight the need for clear guidelines and consent protocols, ensuring that our technology is used responsibly and respects individual privacy and rights.

References

1. Otberdout, N., Ferrari, C., Daoudi, M., Berretti, S., Bimbo, A.D.: Sparse to dense dynamic 3d facial expression generation (2022)
2. Otberdout, N., Ferrari, C., Daoudi, M., Berretti, S., Del Bimbo, A.: Generating multiple 4d expression transitions by learning face landmark trajectories. IEEE Transactions on Affective Computing (2023)
3. Potamias, R.A., Zheng, J., Ploumpis, S., Bouritsas, G., Ververas, E., Zafeiriou, S.: Learning to generate customized dynamic 3d facial expressions (2020)
4. Ranjan, A., Bolkart, T., Sanyal, S., Black, M.J.: Generating 3d faces using convolutional mesh autoencoders. In: Proceedings of the European conference on computer vision (ECCV). pp. 704–720 (2018)
5. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
6. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models (2022)