








GANtitz: Ultra High Resolution Generative Model for Multi-Modal Face Textures

A. Gruber^{1,2}  E. Collins² A. Meka²  F. Mueller²  K. Sarkar² 
S. Orts-Escolano²  L. Prasso² J. Busch² M. Gross¹  T. Beeler² 

¹ETH Zurich, Switzerland

²Google

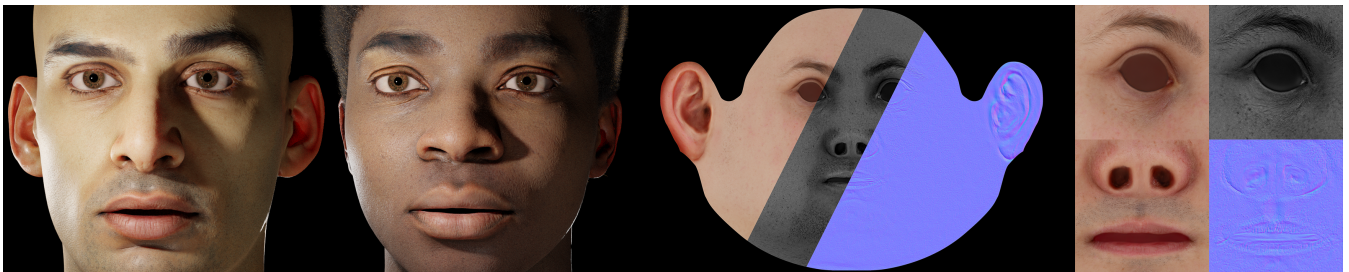


Figure 1: GANtitz generates ultra high resolution multi-modal appearance maps: albedo, specular attenuation, and displacements/normals. These maps can be used to render diverse digital humans in traditional rendering pipelines.

Abstract

High-resolution texture maps are essential to render photoreal digital humans for visual effects or to generate data for machine learning. The acquisition of high resolution assets at scale is cumbersome, it involves enrolling a large number of human subjects, using expensive multi-view camera setups, and significant manual artistic effort to align the textures. To alleviate these problems, we introduce GANtitz (A play on the german noun Antlitz, meaning face), a generative model that can synthesize multi-modal ultra-high-resolution face appearance maps for novel identities. Our method solves three distinct challenges: 1) unavailability of a very large data corpus generally required for training generative models, 2) memory and computational limitations of training a GAN at ultra-high resolutions, and 3) consistency of appearance features such as skin color, pores and wrinkles in high-resolution textures across different modalities. We introduce dual-style blocks, an extension to the style blocks of the StyleGAN2 architecture, which improve multi-modal synthesis. Our patch-based architecture is trained only on image patches obtained from a small set of face textures (<100) and yet allows us to generate seamless appearance maps of novel identities at $6k \times 4k$ resolution. Extensive qualitative and quantitative evaluations and baseline comparisons show the efficacy of our proposed system. (see <https://www.acm.org/publications/class-2012>)

CCS Concepts

• **Computing methodologies** → **Machine learning; Texturing;**

1. Introduction

Capturing high-quality multi-model texture maps from real humans is a challenging task, involving multi-view camera setups, carefully designed illumination, geometry and appearance reconstruction, as well as manual topology alignment and clean-up. Collecting a large-scale dataset is therefore a very expensive and logistically complex endeavor, further complicated by privacy implications.

We present a generative model that can synthesize texture maps of novel identities at very high resolution (6144×4096) from very few training samples (<100). The generated textures include albedo, specularity, and displacements maps from which normals can be computed. Advances in generative adversarial networks, such as StyleGAN [KLA18], allow for generating visually rich imagery in specific domains such as human faces. However, the task of generating high-resolution textures comes with two unique challenges not addressed by state-of-the-art models.

The first challenge is the lack of availability of high-resolution texture maps at a large scale. Training diffusion or generative models, such as StyleGANs [KLA18; KLA*19; KAL*21], requires tens of thousands of images of a given domain. Fewer images lead to either divergent training or mode collapse, severely restricting the diversity of generated images [BGA*22]. We address the severe data scarcity using various discriminator augmentations, similar to [KAH*20].

The second challenge is that representing intricate appearance details requires extremely high resolution. State-of-the-art models already push the limits of compute and memory on modern hardware. StyleGANv3 [KAL*21] takes over 3 months to train on an NVIDIA V100 GPU with 16GB memory at 1K image resolution. For 4K, a 16-fold increase would thus make training prohibitively slow. While the compute requirements may be satisfied by training on multiple GPUs in parallel, memory limitations prevent the processing of batches of high-resolution training images - GPUs with roughly 1TB of memory would be required.

Our method provides a solution to both problems, as it is trained on patches gathered from a small number training texture maps, and yet generates diverse, seamless and full $6k \times 4k$ resolution textures. Texture maps have a canonical UV domain that maps to a defined mesh topology – our architecture makes use of this spatial alignment. Our training dataset consists of aligned facial textures. During training, we discretize the 2D UV domain into a fixed set of overlapping patches. For a given patch, our convolutional generator, conditioned on the desired modality, learns to synthesize a corresponding in-distribution texture patch. The generator is guided by a discriminator that is provided with a per-patch positional encoding, along with the real or generated multi-modal texture. The discriminator thus informs our model about the correctness of the local statistics of the texture at the discriminated UV location.

The different modalities of our generated samples are produced one at a time, using a mechanism we call dual-style blocks. In addition to the latent code that a GAN typically maps to a sample, our generator is further conditioned on a second, modality-specific latent embedding. The generator is thus invoked once per modality, each time with the appropriate embedding.

Our pipeline produces high-quality realistic texture patches, and a seamless and well-composed global texture at full-resolution, avoiding mode-collapse and overfitting. By projecting textures into the learnt latent space, we demonstrate two practical capabilities of our method, namely *modality completion* and *super resolution*.

2. Related Work

2.1. Generative Models of Human Faces

Morphable models of human faces synthesize both geometry and appearance [BV99; PKA*09; LBB*17; YTB*21; BTS*21], but are limited in resolution and high-frequency details due to their inherently low-dimensional representation. [BLC*21] rely on a regional morphable model for an initial fit of multi modal textures, and subsequently training a refiner with an adversarial training scheme.†

Several (Variational) Autoencoder based systems have been proposed [AWB18; BWS*18; CBGB20; BML*21], however these

struggle to produce realistic high frequency detail. [CBGB20] resolve this issue in part by learning a separate super-resolution step. [LSSS18] demonstrate high quality, however their model is specific to a single subject.

StyleGAN models [KALL17; KLA18; KLA*19; KAL*21] have excelled at learning a distribution of real human faces, capable of synthesizing diverse subjects at 1024×1024 resolution. Due to the limits of current hardware, however, StyleGAN cannot be easily scaled up to very high resolutions.

A number of models combine StyleGAN with a traditional morphable model for geometry and expression synthesis. [GPKZ19] and [SSK19] use ProgGAN to synthesize albedo textures. TBGAN [GLP*20] generates albedo and additional modalities using a GAN with modality-specific output branches and [LNK*21] discuss a GAN-based two stage facial digitization framework. [DTP23] train a shape conditioned texture generator supervised by multiple discriminators at different resolutions. [CWZ*21] uses StyleGAN2 to inpaint and render photo-realistic albedo maps. [TEB*20a; TEB*20b] achieve portrait image editing by rigging the StyleGAN latent space using a morphable model fit. [LBZ*20] learn physically based face models from facial scans. [LMP*23] use a generator with a per-modality head, which is then applied to face reconstruction. However the resolution of all of these models is limited to that of the backbone StyleGAN.

AvatarMe [LMG*20; LMP*21] matches our resolution target, similarly generating textures at 6144×4096 . Unlike our method, the generative part produces textures 8 times smaller followed by a super-resolution network. Our method directly renders at the output resolution, which allows it to generate photorealistic high-frequency details. More recently, methods based on diffusion models [HJA20] have shown very promising results for facial assets, such as [ZQL*23; PLMZ23]. Additionally, [ZQL*23] leverage a super-resolution stage to obtain 4096×4096 textures from ones initially generated at 512×512 .

Finally, a novel style-modulation scheme has been proposed for 2D facial stylizing by [SLZC22].

2.2. High-Resolution Generative Models

[SSE21] proposed a technique to generate continuous and seamless infinite panoramas, by aligning latent and image space. [SIE21] devise a method that generates weights to an MLP, which, if evaluated at image coordinates, produces pixel values and subsequently a continuous image. [FAW19] produce large textures by generating tiles with a pretrained generator and then repeatedly replacing tiles with better matches. [ERO20] use a CNN to learn a code book of texture details that are sampled by a transformer network to create images of arbitrary resolution. [YOC*22] proposes a model that can generate high-resolution non-repetitive images from a single training sample. While they generate coherent images, their technique cannot generalize to novel face appearance characteristics such as skin color, hair and other details.

[XWC*20] found that CNNs make use the zero-padded convolutions as an implicit positional encoding. This has led to several architectures being proposed that discard zero-padding in

favor of explicit positional encodings, thereby alleviating aliasing [KAL*21] and facilitating rendering at arbitrary resolutions [LCL*22; CGS*22; NSK*22; DNR*23]. These models can, in theory, be scaled to very high resolution datasets such as ours and additionally allow seamless tiling. We show a baseline comparison with [CGS*22] in Section 5.3.3. [ZHD*22] achieve tileability and scalability as well, but they employ a conditioning input texture which is useful in their context of material generation. Also in the context of material generation, [VMR*23] introduce a method to generate high-resolution, tileable textures with diffusion models through noise rolling. Patch based generation to alleviate memory limitations has also been employed before, more recently in 3D generative models [SLNG21; STWW22]

2.3. Training GANs with Limited Data

Our training data consists of only 97 high resolution face textures, many orders of magnitude less than typically used. Data scarcity can lead to overfitting and instability during training [WRSJ19; KAH*20; ZLL*20].

Some recent methods use regularization to improve the generative capacity either by stabilizing training dynamics [RLNH17; SCT*17; GAA*17; MKKY18; ZK18; JF19], or by preventing mode-collapse [SVR*17; LZWS19; YHJ*19; MLT*19], thereby ensuring diversity in the generative distribution. Loss regularization has been suggested for specifically targeting the data scarcity issue [TIL*21].

We follow another recent line of work that proposes to alleviate this issue by applying data augmentations to the discriminator input to artificially expand the data seen by the discriminator [ZSL*21; TTN*21; ZLL*20]. Some works additionally suggest to adapt the rate at which augmentations are applied according to the performance of the discriminator [KAH*20; JDWL21]. We achieved best performance with a combination of adaptive and non-adaptive augmentations as described in Section 4.2.

3. Dataset

High-resolution face textures that capture intricate appearance details are challenging to acquire for technological, logistical and privacy-related reasons. In this paper, we use commercially available texture maps from www.3dscanstore.com, which consist of a total of 97 aligned 8K face textures captured from distinct subjects.

For each subject, multiple modalities are provided, namely albedo, specular and normal maps. The geometry is obtained from high resolution photogrammetry scans, then retopologized, UV mapped and textured by graphics artists. To remove the hair, stubble is inpainted on the scalp region for subjects who are not bald. Most samples have spatially identical hair stubble distribution on the scalp, but with appearance adjusted to match the subject's skin tone. The resulting 8k resolution textures are aligned, share a common UV domain and can be mapped to different face geometries. The dataset contains 47 female and 50 male subjects.

The artistic preprocessing was performed by the store from which the dataset was acquired. To ensure optimal training, we further apply the following pre-processing to the data:



Figure 2: The albedo map of one sample in our dataset. From left to right: Full 8k map, cropped 6k × 4k map & cropped map with mask, and a head geometry with the result applied.

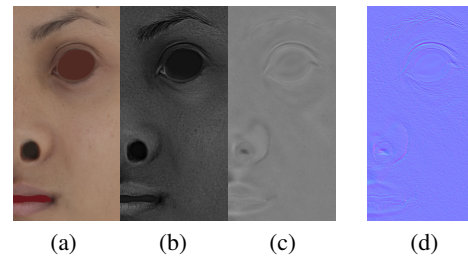


Figure 3: A close up of the different modalities we train on for one sample in our dataset: (a) Albedo, (b) specular intensity, (c) displacement, (d) The normal map from which the displacement map was obtained through integration.

Cropping and masking. A Disproportionately large area of the textures is dedicated to features of relatively low relevance, such as the back and top of the head and the neck. Additionally, the scalp is artistically in-painted and does not contain meaningful variation. We therefore crop the textures from their original 8k resolution down to 6k × 4k, thereby omitting these regions. The resulting image frames the central facial features.

To envelop the relevant region even more precisely, we apply a mask as depicted in Figure 2. Later in training we sample only patches that overlap at least partially with the visible region.

Generating Displacement Maps. We find it beneficial to train for displacement maps rather than the normal maps. Correct normal maps can be computed directly from the displacement maps. To that end, we convert the normal maps in our dataset to displacement maps. We do this via deconvolution with gradient kernels in the spectral domain. The assumed gradient kernel is chosen such that the dynamic range of the subject with most variation remains bounded between 0 and 1. See Figure 3 for examples.

Exploiting Symmetry. By making the assumption that the distribution of the left and right sides of faces are identical up to reflection, we can naturally double our training data by horizontally flipping every sample.

4. Method

Our method builds on the adversarial architecture of StyleGAN2 [KLA*19]. While StyleGAN has been demonstrated to achieve ex-

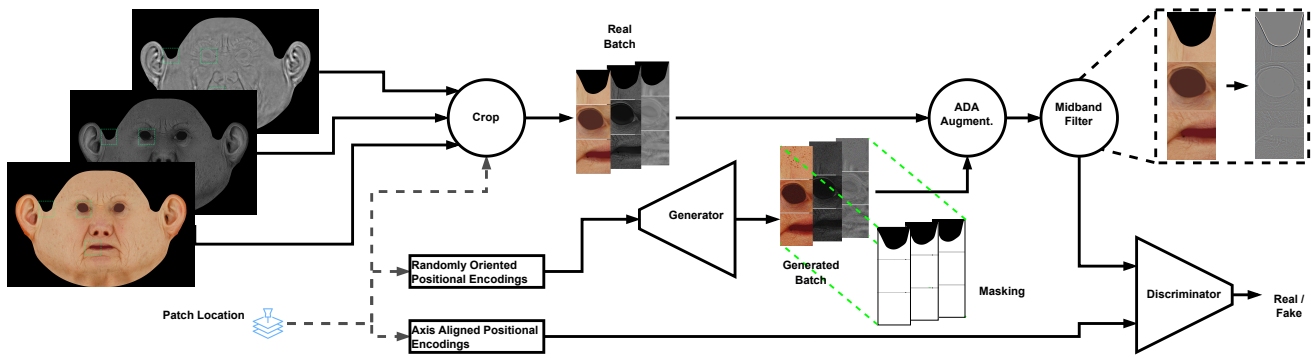


Figure 4: During training we sample overlapping patches of 512×512 from our data set. We prompt our generator to produce patches for the same locations. Due to the aligned nature of our UV textures training-set, we can facilitate the generator’s task by multiplying the UV layout mask onto the generated samples. These real and fake samples are augmented before being fed to the discriminator. While we apply several augmentations at an adaptive rate based on the discriminator performance in line with prior work, we additionally apply a frequency mid-band filter with random frequency bands at a constant rate. This augmentation directs the discriminator’s attention equally at all frequencies and yields a more frequency diverse generator.

cellent image quality, it does not naively scale to our target high resolution and is prone to overfitting in the limited data regime. The primary challenges before our goal are then a) hardware memory constraints due to processing high-resolution images in every training iteration, and b) a severely limited training set size.

To that end we make several updates to the baseline architecture of StyleGAN2 as discussed in Section 4.1. We augment the generated and ground truth images before passing to the discriminator to allow for a more balanced convergence race between the generator and discriminator as conveyed in Sections 4.2 and 4.3. Our resulting generative model can be used to project a novel unseen texture into its latent- and noise space, which allows for unique applications as presented in Section 5.

4.1. Generator

We adapt the StyleGAN2 [KLA*19] architecture since it is still state-of-the-art for high resolution image synthesis, with several improvements proposed in prior work implemented on top of it (The more recent StyleGAN3 is particularly designed to ensure translational and rotational equivariance, which is not relevant to our use case since texture maps are spatially well aligned. We therefore chose to use StyleGAN2 due to its faster runtime, lower memory footprint and comparable image quality and diversity): As described by [XWC*20], we remove the zero padding in our convolutional layers and replace the learned constant with sinusoidal positional encodings. In-line with findings by [KAH*20], we reduce the size of our mapping network to two layers only.

This architecture, however, cannot be trivially scaled from its baseline 1024×1024 resolution to 6144×4096 as modern GPUs cannot accommodate such high resolutions for training. We therefore adopt a patch based approach. While it has been previously demonstrated that for a GAN to learn the synthesis of high frequency detail, it is sufficient to apply adversarial supervision to

local patches [IZZE17], it is not obvious that a training scheme relying solely on patch level supervision would result in a globally coherent image. We show experimentally that for the domain of aligned textures such coherence is indeed attained, and further discuss global consistency in Section 5.1.1. Naively tiling patches will cause seams in the resulting images, however padding free convolutional layers in conjunction with pixel aligned per-patch-location positional encoding yield seamless patches by construction. We provide further details on considerations for the receptive field and positional encoding computations in the supplementary document.

4.1.1. Multi-Modality and Dual-Styles

The natural approach to generating multiple modalities is a modification to the number of output and input channels in the generator and the discriminator, respectively. However, we found this to be particularly demanding for the generator. As a result, much more reliance on augmentations to stabilize training is required which we found to be harmful to the training over all. See the ablation in section 5.6.6 for a comparison.

Instead, we equip our generator with a mechanism we dub dual-style blocks, shown in Figure 5. In this setting, the generator is conditioned on the desired modality, and the generation of multi-modal samples is done over successive evaluations, one per modality. The conditioning input is a per-modality learned embedding. These embeddings are utilized as additional style inputs and learned bias offsets for each style block in convolutional layers. As specularities and displacement maps are single channel, we average the three output channels of the generator after synthesis for those two modalities. Normals can then be derived from the displacement map.

4.1.2. Output Masking

We facilitate the generative process further by explicitly applying the region mask over the generated output, prior to discrimination.

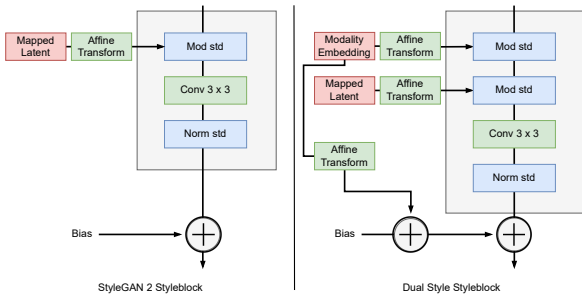


Figure 5: On the left is a depiction of the style block as proposed in [KLA*19]. Note that we visualize the modulation and demodulation as acting on the feature map and not the weights. This is purely for clarity and can be implemented as weight modulation for efficiency. On the right is a diagram of our proposed dual-style block: From the learned per-modality embedding we derive an additional style as well as a bias offset. We thus have an "outer" style (i.e., the modality style) and an "inner" style (the identity style). The derived bias is added to the standard learned bias.

Given the patch location we simply retrieve the relevant part of the mask and perform a multiplication. The generator then no longer needs to learn the otherwise stark discontinuities and it additionally serves as a positional inductive bias as has been shown by [XWC*20] to be beneficial.

4.2. Discriminator Augmentation Scheme

With only 97 samples at our disposal, we operate in a setting of severe data scarcity that must be addressed to ensure stable training. We employ discriminator augmentations as proposed by [KAH*20]. These augmentations are differentiable transformations applied to both the real and generated data before being fed to the discriminator. Each transformation is employed probabilistically at a rate p . It is suggested that when applied at $p < 0.85$, these augmentations do not "leak" into the generator and therefore do not negatively impact the learned data distribution. Moreover, the rate is adaptively chosen based on the discriminator's performance so as to balance the two adversaries. These augmentations comprise the addition of stochastic noise, partial spatial masking and 2D-transformations, namely rotation, scale and translation. We found color transformations to be particularly leaky in our setting and thus avoid them.

We do not apply all augmentations at an adaptive rate. We apply a mid-band filter augmentation at a constant rate of $p = 0.85$, implemented as a convolution with a difference of two Gaussians with randomly sampled variances. The ablation in Section 5.6.2 conveys the benefit over adaptive application. The mid-band filter can be interpreted as a mechanism to direct attention, forcing the discriminator to consider the entire frequency spectrum equally. It thus results in a more frequency diverse generator, which is particularly apparent in generated high frequency detail such as wrinkles and stubble as depicted in Figure 6. In Section 4.3.1, we introduce another non-adaptive augmentation we apply to the small variant of



Figure 6: Bottom: When trained with adaptive frequency augmentations. Top: Our method. Note the much better definition of intricate structures such as the wrinkles on the lips.

our model. These non-adaptive augmentations already ensure stable convergence for the better duration of the entire training.

4.3. Discriminator Architecture

Our discriminator architecture mostly follows [KLA*19]. We differ only in that we facilitate localization by concatenating per-patch axis-aligned positional encodings to the samples before providing them to the discriminator. Please refer to the supplementary document for details on their computation.

We follow [KLA18] in using a standard deviation layer to promote diversity, computed across the batch dimension. We estimate the deviation across samples from the *same patch location*, as otherwise the majority of the diversity is location dependent rather than identity related. As such, we arrange batches in groups of successive samples from the same location.

4.3.1. Memory Constraints and Modality Dropout

When training on A100 GPUs, the memory budget is sufficient to train a large version of our model which we call **GANtLitz High Capacity Model (GANtLitz-L)**. When training on V100 GPUs, the reduced memory budget necessitates some adjustments in model architecture and training procedure, which we call **GANtLitz Standard Capacity Model (GANtLitz-S)**.

The total memory requirements for a single training iteration depend on model size and the number of modalities to be generated. Since model size affects the generator's ability to learn the distribution, a smaller model producing less diverse and detailed results, for **GANtLitz-S** we reduce the number of parameters in the final two layers only, halving the number of filters compared to **GANtLitz-L**.

While training on each modality in isolation further reduces the memory footprint, in order for cross-modality coherence to be enforced by **GANtLitz-S** we must nonetheless train on multiple modalities at once. We find a compromise by generating two modalities at a time, dropping-out the third (substituting it with zeros), and likewise masking one modality per sample on real data as well.

This scheme can be interpreted as a discriminator augmentation and we discuss it in more detail in Section 4.2. An ablation with respect to a training scheme where a lower capacity model avoids the necessity for modality dropout is given in Section 5.6.3.

← Layer res.	48	96	192	384	768	1536	3K	6K
← Patch res.	4	8	16	32	64	128	256	512
← Patch stride.	2	4	8	16	32	64	128	256
← Rec. field of single pixel	10	6	6	6	6	5	4	1
← Rec. field of patch backward	14	14	22	38	70	133	259	512
→ Padded Patch size forward	14	14	22	38	70	134	262	518

Table 1: Starting with a single pixel and a 512 patch at the output $6k \times 4k$ resolution, the third and fourth rows respectively give the size of the corresponding receptive field at every resolution block. Note that due to rounding-up sub-pixel receptive fields, the forward pass generates a larger patch, and that the first level is special as it contains no up-sampling and blurring layer.

4.4. Implementation Details

4.4.1. Patch-wise Generator

We partition the desired 6144×4096 output textures into 23×15 overlapping patches with 50% overlap. At test time, we synthesize 12×8 non-overlapping and seamless patches that are then stitched together into one coherent full resolution texture. The patch resolution therefore amounts to 512×512 .

Figure 7 visualizes our generator architecture. The feature map pyramid, starting from patch specific positional encodings at a resolution of 14×14 and 512 channels, successively increases the feature map resolutions across 8 levels, finally yielding a $518 \times 518 \times 32$ feature map. As proposed by [KLA*19], each level additionally produces a RGB output, and all these outputs are composed into the final generator output. To obtain the final patch we crop that output to the central 512×512 area.

4.4.2. Choice of Positional Encodings

For illustrative purposes in Figure 7 we suggest the initial tensor of positional encodings to be the result of a crop operation applied to a larger 42×58 coordinate map. In practice the per-patch encodings can be computed on the fly given the patch location. To ensure seamless patch generation we take care that the receptive fields of neighboring patches, to the degree that they overlap, are identical. Conceptually, this can be accomplished by having the initial encodings be cropped with precise pixel alignment. Two neighboring, overlapping patches that are a stride of 256 pixels apart in the output thus correspond to two neighboring patches in the input that are a stride of 2 pixels apart.

This suggests that to sample the desired number of patches we only need 32×48 pixels in the initial map. To accommodate for the receptive field of the generator, avoiding zero padding, we extend the coordinate tensor by an amount corresponding to the padding needed based on the generator’s receptive field.

4.4.3. Generator Receptive Field

Figure 8 conveys how the receptive field develops when traced through a single resolution level in the generator. The generator can

be divided into two paths: The feature map pyramid and the `to_rgb` branch. We can similarly trace a pixel or a whole patch through the entire generator as presented in Table 1. For the required 512×512 output resolution the patch resolution and the receptive field at the initial layer amount to 4×4 and 14×14 , respectively. The per-side padding thus amounts to $\frac{(14-4)}{2} = 5$. As a result, the coordinate tensor measures 42×58 .

4.4.4. Discriminator Patch-Location Conditioning

To facilitate the patch localization in the discriminator we condition it on sinusoidal encodings as follows: The highest frequency is chosen to have a period of two times the side-length of a patch, i.e. 1024 pixels, and that of the lowest one equivalent to twice the longer side-length of the full texture, i.e. 12288 pixels. In between we desire a drop in frequency by a factor of 2 between channels. Overall the period drops by a factor of $12288/1024 = 12$ and we therefore require $\log_2 12 = 3.25 \approx 4$ steps, resulting in a total of 5 channels. In line with prior work we compute such encodings for sine and cosine, as well as for both spatial dimensions, a total of 20 channels. Thus, assuming we train for three modalities (albedo, specular and displacement maps) with 3, 1 and 1 channels, respectively, we end up with 25 channels in the input layer of the discriminator.

4.4.5. Training Parameters

We use the Keras implementation of the Adam Optimizer with $\beta_1 = 0$, $\beta_2 = 0.99$ and a target learning rate of 0.003, which we linearly ramp up from 0 over the first 2000 iterations. We apply R1 regularization with a weight of 2.

5. Experiments

We demonstrate the efficacy of our method in four parts: By *Qualitative Analysis*, by proposing applications based on *Inversion*, through *Baseline Comparisons* and finally with *Ablation Studies*. Finally, we also explore the model’s behavior when extrapolated into the untrained regions of the canvas.

We conducted most experiments, including the ablation study, using NVIDIA V100 GPUs with the **GANtLitz-S** configuration of our method and a per-GPU batch size of 3. We also used A100 GPUs which, by virtue of higher memory capacity, enabled us to train the **GANtLitz-L** variant and a per-GPU batch size of 4.

5.1. Qualitative Analysis

We present qualitative results produced by our method in Figure 9. The generated samples exhibit excellent coherence across the different appearance maps. Notably, the facial details are rendered with high resolution, clearly depicting skin pores and stubble. Figure 11 demonstrates the smooth transition between different facial appearances achieved through latent space interpolation. Moreover, Figure 10 showcases the ability to obtain variations in fine-scale features such as pores and wrinkles through noise re-sampling.

To better convey the visual fidelity our method exhibits, we strongly encourage the reader to engage with the accompanying video. Additionally, in our supplementary document, we provide

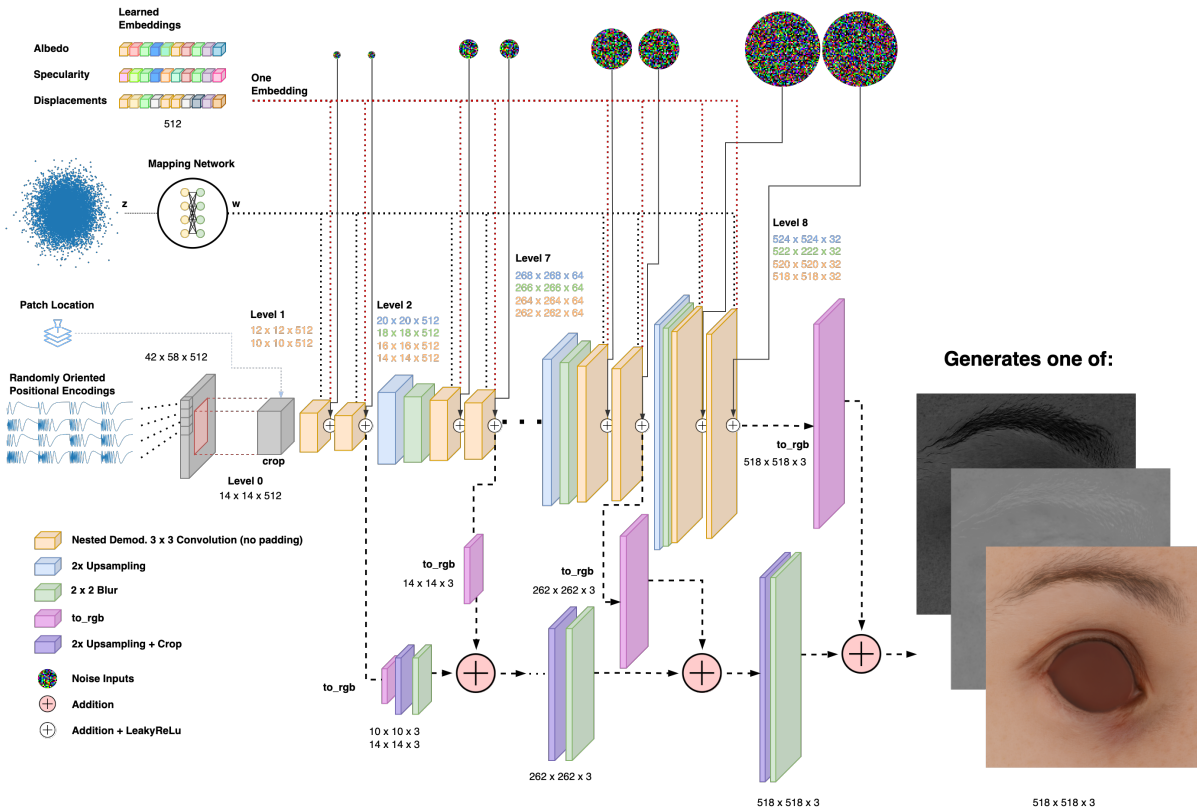


Figure 7: Starting from patch specific positional encodings, our synthesis network produces a seamless patch by sequentially applying upsampling and padding-free demodulated convolution layers, among others. Sequential synthesis of all non-overlapping patches yields the full $6k \times 4k$ resolution image.

further generated samples at both the global and patch levels, as well as comparative figures of training samples. All results were generated from our **GANtLitz-L** model with an exponentially moving average on the model weights with a decay factor of 0.999.

5.1.1. Global and Local Characteristics

Despite the absence of explicit global supervision, our generated samples exhibit good global structure as shown by the top row of **Figure 9** and many more examples in our supplementary document. Due to the limited training dataset, low-frequency variation is limited. This finding is further confirmed by noise sampling experiments, presented in **Figure 10**. Low frequency structure remains largely unchanged while mid- and high-frequency details show good variation.

5.2. Inversion

We can project textures into the latent space of the generator. We optimize for the per-layer latent code and noise maps in two stages: First, only the latent codes are optimized while the noise maps are re-sampled at each optimization iteration. In the second stage, the latent codes are fixed while the noise maps are fine-tuned. Similar to [KLA*19], we regularize the noise using a pixel correlation penalty. We also penalize the mean and standard deviation of the

noise distribution. Inversion then allows for two intriguing resulting applications: *Modality Completion* and *Super Resolution*.

5.2.1. Modality Completion

Given texture maps for a single modality, e.g. albedo, we can estimate corresponding other modalities, i.e. specularity and displacement. As our generator generates one modality at a time, we invert the target albedo by feeding the albedo modality embedding to the generator’s dual-style blocks. After optimizing for the latent code for the target albedo, we use the specularity or displacement embeddings to generate the corresponding maps. **Figure 13** shows the result of inversion and modality completion. Our method faithfully reconstructs the target albedo, down to freckles and stubles, and estimates plausible specular and displacement maps. This is particularly useful as acquiring specular and displacement assets is cumbersome whereas albedo acquisition is comparatively easier by employing a cross polarized setup.

5.2.2. Super-resolution

Our method also allows us to invert low resolution textures, which enables super-resolution to the full $6k \times 4k$ resolution with stochastic high-frequency details. The reconstruction loss is computed after a downsampling step. During inversion, special care must be

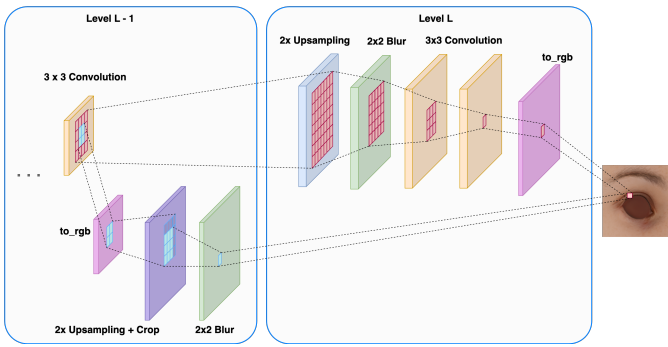


Figure 8: The receptive field of an output pixel is traced back through various network layers. Sub-pixel sizes are rounded up to the nearest integer. As a result, a larger output patch is synthesised during a forward pass, which we then crop down to the desired size. The receptive field modifications occurring in the feature map pyramid (visualized as red pixels) dominate the modifications occurring in the `to_rgb` branch (blue pixels). Tracing an individual pixel’s receptive field backwards we get: 1 (before `to_rgb`), 3 (before `conv`), 5 (before `conv`), 7 (before `blur`), $\lceil 3.5 \rceil = 4$ (before `upsampling`).

taken for noise regularization; the many-to-one correspondence between generated and ground truth pixels can result in degenerated values. To that end we regularize high-resolution noise maps more than low-resolution ones. The combination of modality completion and super-resolution enables the conversion of existing low-resolution assets into high quality texture assets, as seen in Figure 12. Note the fine-scale details synthesized by our method, for example in the region below the eye.

5.3. Baseline Comparisons

We compare against three baseline methods: A global **StyleGAN V2 (StyleGAN2)** model ([KLA*19]) at a resolution of 1024×1024 , **AvatarMe++** [LMG*20; LMP*21], and **Anyresolution GAN (AnyresGAN)** ([CGS*22]). The first comparison underlines the challenge of data scarcity even at low resolution, while the other two highlight the difficulty of high resolution synthesis.

5.3.1. StyleGAN V2

StyleGAN2 is typically trained on tens of thousands of samples, we therefore again employ adaptive discriminator augmentations [KAH*20]. To provide a meaningful comparison, we adapt StyleGAN2 to produce all three modalities, resulting in 5 output channels. Moreover, we multiply the generator output with the UV layout mask as with our own method. We refer to this resulting model as **Multi-Modal StyleGAN V2 with ADA (MMStyleGAN2-ADA)**.

Figure 14 compares **MMStyleGAN2-ADA** with our method and a real sample as reference, the latter two down-sampled to $1K$ resolution. We qualitatively observe that **MMStyleGAN2-ADA** fails to reproduce fine details. While quantitative diversity evaluation, discussed in subsection 5.5, indicates a faithful distribution reproduction, the model overfits to the training data leading to a discon-

tinuous latent space. We encourage the reader to consult the complementing video.

5.3.2. AvatarMe++

[LMP*21] propose a method capable of producing multi-modal texture maps at the same resolution as our method; more specifically, an initial texture is sampled from the GAN of [GPKZ19]. The pipeline proposed by [LMP*21] is then used to upscale the textures to the final size and to infer the missing modalities. Figure 15 shows a direct comparison between the two methods. While both methods produce the same resolution, a clear difference in the definition of high frequency details, such as wrinkles, pores and stubble can be observed. It must be noted that [LMP*21] was not designed to be a generative model but rather a framework for the construction of avatars from in-the-wild images.

5.3.3. Anyres GAN

AnyresGAN aims to learn a dataset at any resolution simultaneously. The method is trained in two phases: A global **StyleGAN V3 at 256×256 (StyleGAN3-256)** ([KAL*21]) teacher network learns the distribution at a fixed low-resolution. In the subsequent multi-resolution training phase, a **StyleGAN3** student network, which we refer to as **AnyresGAN**, is conditioned on location- and scale-specific positional encodings and learns to synthesize a patch. The loss consists of an adversarial loss and a reconstruction loss with respect to the output of the teacher model.

Training the teacher network on our dataset proved to be challenging: At low-resolution and with few training samples the model quickly diverged. We therefore opted for a **StyleGAN V3 at 256×256 with ADA (StyleGAN3-256-ADA)** model instead. While divergence was averted, the high rate of augmentation severely impacted the learned distribution. Applying our UV layout mask to the samples post synthesis alleviated these issues substantially. Nonetheless, the resulting samples exhibit artifacts, as is also confirmed by our quantitative analysis. See Figure 17.

The multi-resolution phase posed similar challenges. Limited data along with the flawed teacher, in conjunction with a large resolution pyramid to absorb, prevented good convergence. Limiting the dataset resolution to $1K \times 1K$ permitted convergence, but high frequency details remain absent. In light of these findings we forewent multi-modal experiments with AnyresGAN.

5.4. Extrapolation

While we limit the area to be learned explicitly with our masking scheme, we can in fact extrapolate the synthesis into the untrained region. See Figure 16. While the initial results demonstrate an inconvenient blotchiness in the lower frequencies, this can be fixed with a strong blur operation applied to the low frequencies of the untrained region only. The generated local skin details even in these areas are plausible albeit generic and uniform.

5.5. Quantitative Evaluation

We analyze the degree to which our method captures the data distribution. As quantitative metric we employ the **Kernel Inception Distance (KID)** as proposed by citemystifyingMMDGans.

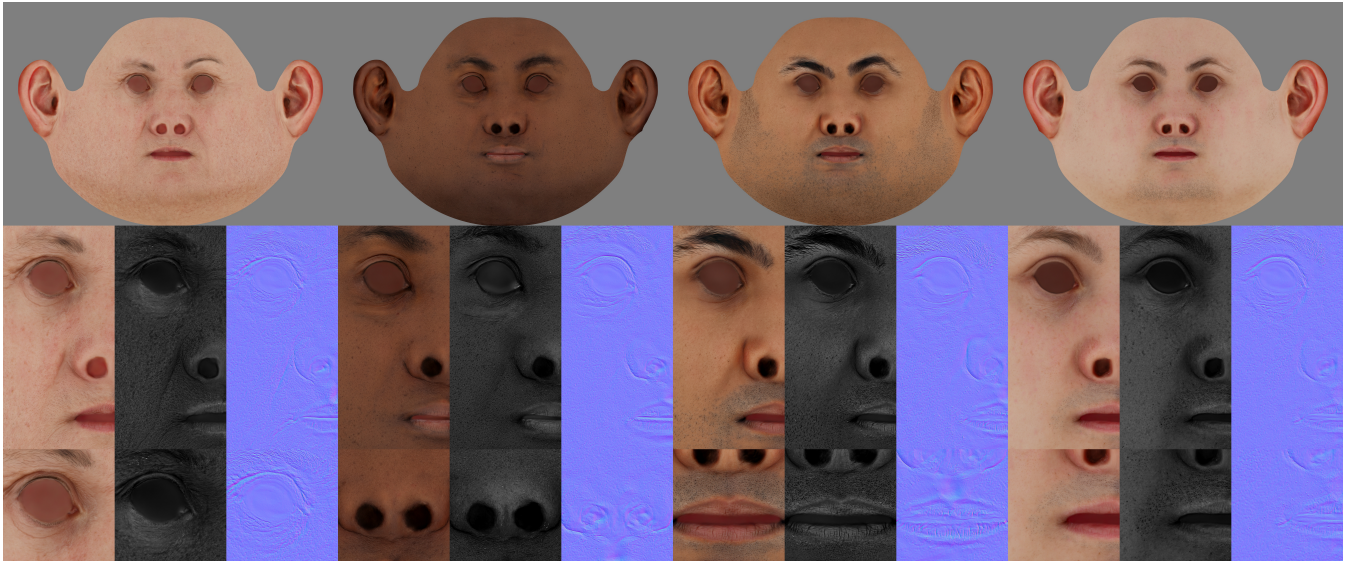


Figure 9: Sampling. Our method allows to unconditionally sample multi-modal appearance maps (albedo/specular attenuation/normals). Here we show four random samples exhibiting different skin complexion and skin details. Please zoom in to appreciate the skin details.

It has much better convergence characteristics in comparison to the **Fréchet Inception Distance (FID)**, and is unbiased, two properties shown to be important when working with very limited data. With limited training data, however, it is difficult to calibrate the resulting KID values, since this is usually done using large (statistically significant) training set splits. As we have only 97 subjects, the splits contain less than 50 samples each and exhibit large variance.

We compute the KID distribution over 1000 random splits of training data. We similarly generate sets of the same size when evaluating the generated distribution. The overlap between distributions in terms of both mean and variance provides insight into the discrepancy between the generated and real data distributions. [Figure 17](#) conveys the KID values of all models we discuss. By design, the **KID** values for training splits are distributed around zero.

The qualitative findings in Section 5.3.3 on the **AnyresGAN** baseline are confirmed numerically as well: The teacher model exhibits weak performance and the second training phase fares no better. **MMStyleGAN2-ADA** on 1k whole images matches the real distribution well, benefiting from being trained not on patches but on the whole image, but overfits to the training data. Our model, despite having only patch-level supervision, nonetheless demonstrates good performance at the global scale as well.

At the patch level we confirm that our **GANTlitz-L** model outperforms the other variants also quantitatively. We now discuss various ablations that empirically support our design choices.

5.6. Ablation Study

In this section we ablate various aspects of our method. Each ablation differs from our **GANTlitz-S** model in one aspect. The identifier for each model is listed in the beginning of each of the follow-

ing sections. The quantitative performance of the different variants are presented in [Figure 17](#) in terms of KID. We refer to these results in all following subsections of the ablation study.

5.6.1. No augmentations

NO-AUG As we show with this ablation, data augmentations are critical for model convergence under data scarcity. Without them, the model fails to converge.

5.6.2. Adaptive frequency-band augmentations

ADAPTIVE-FREQ Our complete model applies the frequency-band augmentation at a constant rate of 85%. If we instead adapt the rate of based on discriminator performance, we see a reduction in learned high frequency detail. See [Figure 6](#) for a side-by-side comparison.

5.6.3. Smaller network with optional modality dropout

SMALL-NO-DROP One of the constant augmentations used by **GANTlitz-S** is modality dropout, i.e., feeding the discriminator a partial set of modalities. Since this augmentation also serves to reduce memory usage, applying it adaptively requires a reduction in model size.

We test this trade-off by halving the last layer of the generator and first two layers of the discriminator and find that performance decreases considerably.

5.6.4. No positional information to discriminator

NO-POS-DISC Convergence of our patch-wise training scheme is not possible without providing positional encodings to the discriminator. Without this extra signal to distinguish each of the 345 possible patch locations, the system fails to discover the global structure.

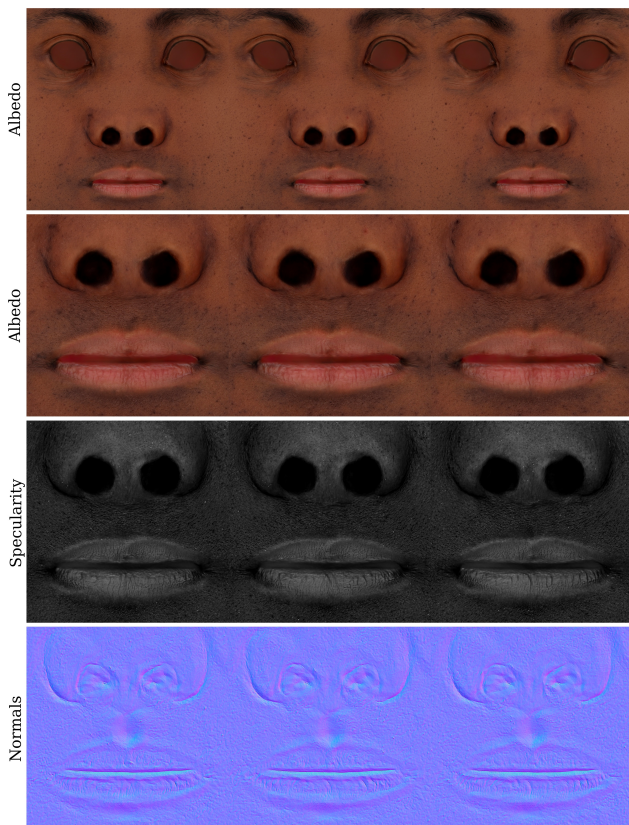


Figure 10: Noise variation with fixed latent code. Texture assets preserve coarse facial appearance but show variation in finer structures like freckles, indicating that our latent space uniquely preserves identity and the noise input is capable of generating variations in fine scale details.

5.6.5. UV coordinates to discriminator

UV-DISC Instead of sinusoidal positional encodings, we feed two-dimensional UV coordinates as positional information to the discriminator. In this case it is harder for the discriminator to use this signal and we observe a degradation in performance.

5.6.6. Multi-channel output instead of dual-style blocks

NO-DUAL A significant architectural choice in our model is the use of per-modality latents in dual-style blocks. In this comparison we see the benefit of this choice compared to a traditional architecture where different modalities are different output channel.

5.6.7. No modality bias

NO-BIAS In this ablation we forego the per-modality bias term in the style blocks. Indeed, the bias term facilitates the modeling of the different modality distributions.

6. Limitations and Future Work

Our method can be improved with the availability of a larger training set. As it is, achieving both high quality and diversity is very

challenging. Data scarcity currently impacts the inversion capability of our model: Many target samples fall out of distribution, causing latent optimization to yield a poor fit. Additionally, reproducing the entire training distribution while retaining a smooth latent space remains challenging. This opens interesting possibilities for future work, such as leveraging larger low- to mid-resolution datasets in addition.

In the spirit of **AnyresGAN**, a global model can be used as a teacher network. However, both the **StyleGAN3-256-ADA** as well as the **MMStyleGAN2-ADA** would not be fit for purpose due to their problems with quality and overfitting, respectively.

The lack of an explicit global supervision mechanism limits the applicability of our method to aligned datasets. Its addition could enable a model to generate super high-resolution images based on non- or partially-aligned data as well.

Acknowledgements

Open access funding provided by Eidgenössische Technische Hochschule Zurich.

References

- [AWB18] ABREVAYA, VICTORIA, WUHRER, STEFANIE, and BOYER, EDMOND. “Multilinear Autoencoder for 3D Face Model Learning”. Mar. 2018, 1–9. DOI: [10.1109/WACV.2018.000072](https://doi.org/10.1109/WACV.2018.000072).
- [BGA*22] BERMANO, AMIT HAIM, GAL, RINON, ALALUF, YUVAL, et al. “State-of-the-Art in the Architecture, Methods and Applications of StyleGAN”. *Computer Graphics Forum* (2022). ISSN: 1467-8659. DOI: [10.1111/cgf.145032](https://doi.org/10.1111/cgf.145032).
- [BLC*21] BAO, LINCHAO, LIN, XIANGKAI, CHEN, YAJING, et al. *High-Fidelity 3D Digital Human Head Creation from RGB-D Selfies*. 2021. arXiv: [2010.05562 \[cs.CV\]](https://arxiv.org/abs/2010.05562) 2.
- [BML*21] BUEHLER, MARCEL C., MEKA, ABHIMITRA, LI, GENGYAN, et al. “VariTex: Variational Neural Face Textures”. *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021 2.
- [BTS*21] B R, MALLIKARJUN, TEWARI, AYUSH, SEIDEL, HANSPETER, et al. “Learning Complete 3D Morphable Face Models from Images and Videos”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021 2.
- [BV99] BLANZ, V. and VETTER, T. “A Morphable Model for the Synthesis of 3D Faces”. *Computer Graphics 33. Annual Conference Series* (1999), 187–194 2.
- [BWS*18] BAGAUTDINOV, TIMUR, WU, CHENGLEI, SARAGIH, JASON, et al. “Modeling Facial Geometry Using Compositional VAEs”. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, 3877–3886. DOI: [10.1109/CVPR.2018.004082](https://doi.org/10.1109/CVPR.2018.004082).
- [CBGB20] CHANDRAN, PRASHANTH, BRADLEY, DEREK, GROSS, MARKUS, and BEELER, THABO. “Semantic Deep Face Models”. *2020 International Conference on 3D Vision (3DV)*. 2020, 345–354. DOI: [10.1109/3DV50981.2020.000442](https://doi.org/10.1109/3DV50981.2020.000442).
- [CGS*22] CHAI, LUCY, GHARBI, MICHAEL, SHECHTMAN, ELI, et al. “Any-resolution training for high-resolution image synthesis.” (2022) 3, 8.
- [CWZ*21] CHANDRAN, PRASHANTH, WINBERG, SEBASTIAN, ZOISS, GASPARD, et al. “Rendering with style: combining traditional and neural approaches for high-quality face rendering”. *ACM Transactions on Graphics (TOG)* 40.6 (2021), 1–14 2.
- [DNR*23] DIOLATZIS, STAVROS, NOVAK, JAN, ROUSSELLE, FABRICE, et al. “MesoGAN: Generative Neural Reflectance Shells”. *Computer Graphics Forum* (2023). URL: <http://www-sop.inria.fr/revues/Basilic/2023/DNRGARD233>.

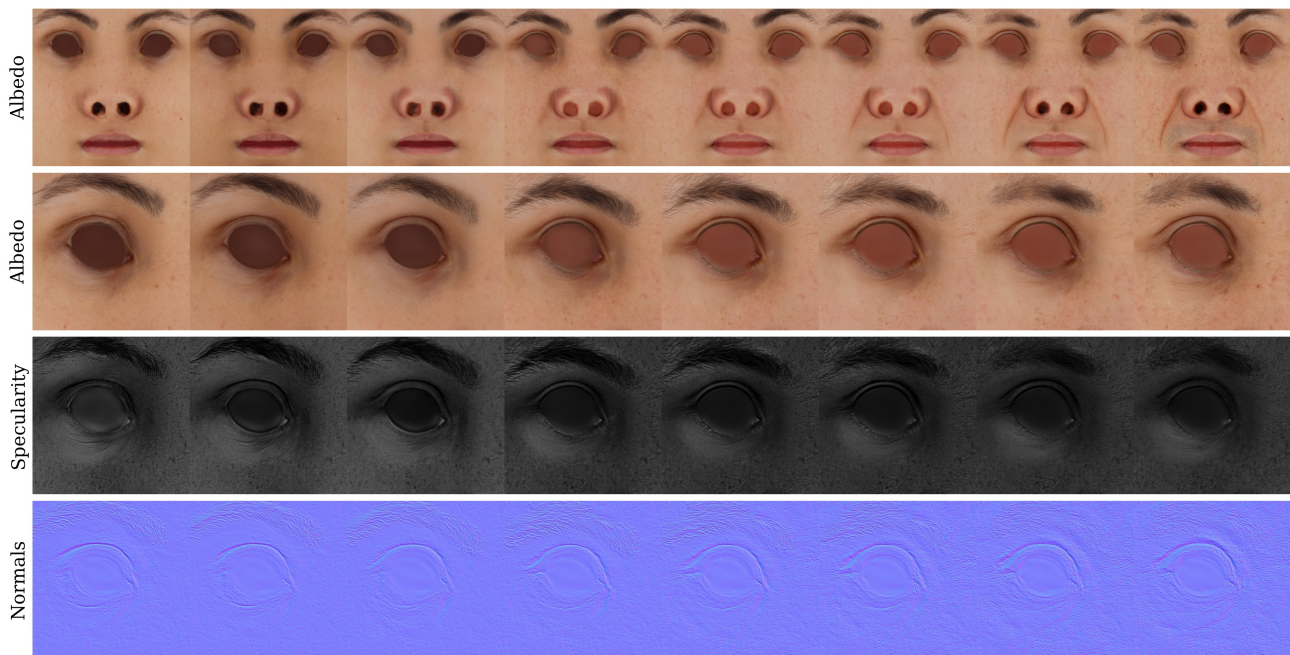


Figure 11: Linear interpolation between two randomly sampled latent codes with intermediate outputs. The results show a smooth transition of spatial features, indicating a continuous latent space of identities. Note how the wrinkles around the eye change gracefully across interpolated frames.

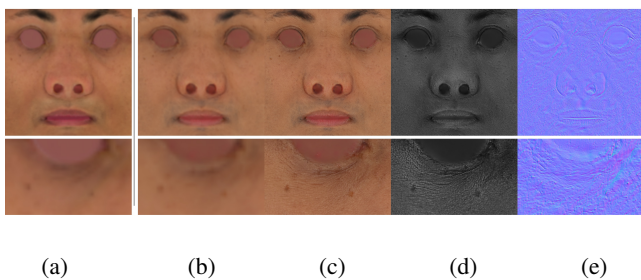


Figure 12: Super Resolution. Inverting an (a) input low-resolution albedo image yields a (b) reconstructed low-resolution albedo as well as the (c) generated high-resolution albedo, (d) specular attenuation, and (e) normals.

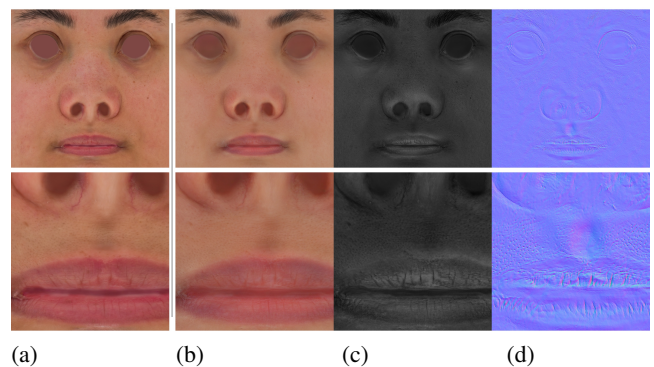


Figure 13: Modality completion. By inverting the (a) albedo texture only our model can generate (b) an albedo texture that closely matches the target texture alongside the other appearance maps: (c) specular attenuation (d) normals. This generates a consistent multi-modal appearance ready to render.

[DTP23] DEB, DEBAYAN, TRIPATHI, SUVIDHA, and PURI, PRANIT. *MUNCH: Modelling Unique 'N Controllable Heads*. 2023. arXiv: 2310.02753 [cs.CV] 2.

[ERO20] ESSER, PATRICK, ROMBACH, ROBIN, and OMMER, BJÖRN. *Taming Transformers for High-Resolution Image Synthesis*. 2020. arXiv: 2012.09841 [cs.CV] 2.

[FAW19] FRÜHSTÜCK, ANNA, ALHASHIM, IBRAHEEM, and WONKA, PETER. “TileGAN: synthesis of large-scale non-homogeneous textures”. *ACM Transactions on Graphics* 38.4 (July 2019), 1–11. ISSN: 1557-7368. DOI: 10.1145/3306346.3322993. URL: <http://dx.doi.org/10.1145/3306346.3322993> 2.

[GAA*17] GULRAJANI, ISHAAN, AHMED, FARUK, ARJOVSKY, MARTIN, et al. “Improved training of wasserstein gans”. *Advances in neural information processing systems* 30 (2017) 3.

[GLP*20] GECER, BARIS, LATTAS, ALEXANDROS, PLOUMPIS, STYLIANOS, et al. “Synthesizing Coupled 3D Face Modalities by Trunk-Branch Generative Adversarial Networks”. *European conference on computer vision*. Springer, 2020, 415–433 2.

[GPKZ19] GECER, BARIS, PLOUMPIS, STYLIANOS, KOTSIA, IRENE, and ZAFEIRIOU, STEFANOS. “GANFIT: Generative Adversarial Network Fitting for High Fidelity 3D Face Reconstruction”. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, 1155–1164 2, 8.



Figure 14: Side-by-side comparison of real data at 1K and 8K resolution, *MMStyleGAN2-ADA*, *AnyresGAN* and our method.

- [HJA20] HO, JONATHAN, JAIN, AJAY, and ABBEEL, PIETER. “Denoising Diffusion Probabilistic Models”. *arXiv e-prints*, arXiv:2006.11239 (June 2020), arXiv:2006.11239. arXiv: [2006.11239](https://arxiv.org/abs/2006.11239) [cs.LG] 2.
- [IZZE17] ISOLA, PHILLIP, ZHU, JUN-YAN, ZHOU, TINGHUI, and EFROS, ALEXEI A. “Image-to-image translation with conditional adversarial networks”. *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, 1125–1134 4.
- [JDWL21] JIANG, LIMING, DAI, BO, WU, WAYNE, and LOY, CHEN CHANGE. “Deceive D: Adaptive pseudo augmentation for gan training with limited data”. *Advances in Neural Information Processing Systems* 34 (2021), 21655–21667 3.
- [JF19] JENNI, SIMON and FAVARO, PAOLO. “On stabilizing generative adversarial training with noise”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, 12145–12153 3.
- [KAH*20] KARRAS, TERO, AITTALA, MIIKA, HELLSTEN, JANNE, et al. “Training Generative Adversarial Networks with Limited Data”. *Proc. NeurIPS*. 2020 2–5, 8.
- [KAL*21] KARRAS, TERO, AITTALA, MIIKA, LAINE, SAMULI, et al. “Alias-Free Generative Adversarial Networks”. *arXiv e-prints*, arXiv:2106.12423 (June 2021), arXiv:2106.12423. arXiv: [2106.12423](https://arxiv.org/abs/2106.12423) [cs.CV] 2, 3, 8.
- [KALL17] KARRAS, TERO, AILA, TIMO, LAINE, SAMULI, and LEHTINEN, JAAKKO. “Progressive Growing of GANs for Improved Quality, Stability, and Variation”. *arXiv e-prints*, arXiv:1710.10196 (Oct. 2017), arXiv:1710.10196. arXiv: [1710.10196](https://arxiv.org/abs/1710.10196) [cs.NE] 2.
- [KLA*19] KARRAS, TERO, LAINE, SAMULI, AITTALA, MIIKA, et al. “Analyzing and Improving the Image Quality of StyleGAN”. *arXiv e-prints*, arXiv:1912.04958 (Dec. 2019), arXiv:1912.04958. arXiv: [1912.04958](https://arxiv.org/abs/1912.04958) [cs.CV] 2–8.
- [KLA18] KARRAS, TERO, LAINE, SAMULI, and AILA, TIMO. “A Style-Based Generator Architecture for Generative Adversarial Networks”. *arXiv e-prints*, arXiv:1812.04948 (Dec. 2018), arXiv:1812.04948. arXiv: [1812.04948](https://arxiv.org/abs/1812.04948) [cs.NE] 1, 2, 5.
- [LBB*17] LI, TIANYE, BOLKART, TIMO, BLACK, MICHAEL. J., et al. “Learning a model of facial shape and expression from 4D scans”. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)* 36.6 (2017), 194:1–194:17. URL: <https://doi.org/10.1145/3130800.3130813> 2.
- [LBZ*20] LI, RUILONG, BLADIN, KALLE, ZHAO, YAJIE, et al. “Learning Formation of Physically-Based Face Attributes”. June 2020, 3407–3416. DOI: [10.1109/CVPR42600.2020.003472](https://doi.org/10.1109/CVPR42600.2020.003472).
- [LCL*22] LIN, CHIEH HUBERT, CHENG, YEN-CHI, LEE, HSIN-YING, et al. “InfinityGAN: Towards Infinite-Pixel Image Synthesis”. *International Conference on Learning Representations*. 2022. URL: <https://openreview.net/forum?id=ufGMqIM0a4b3>.

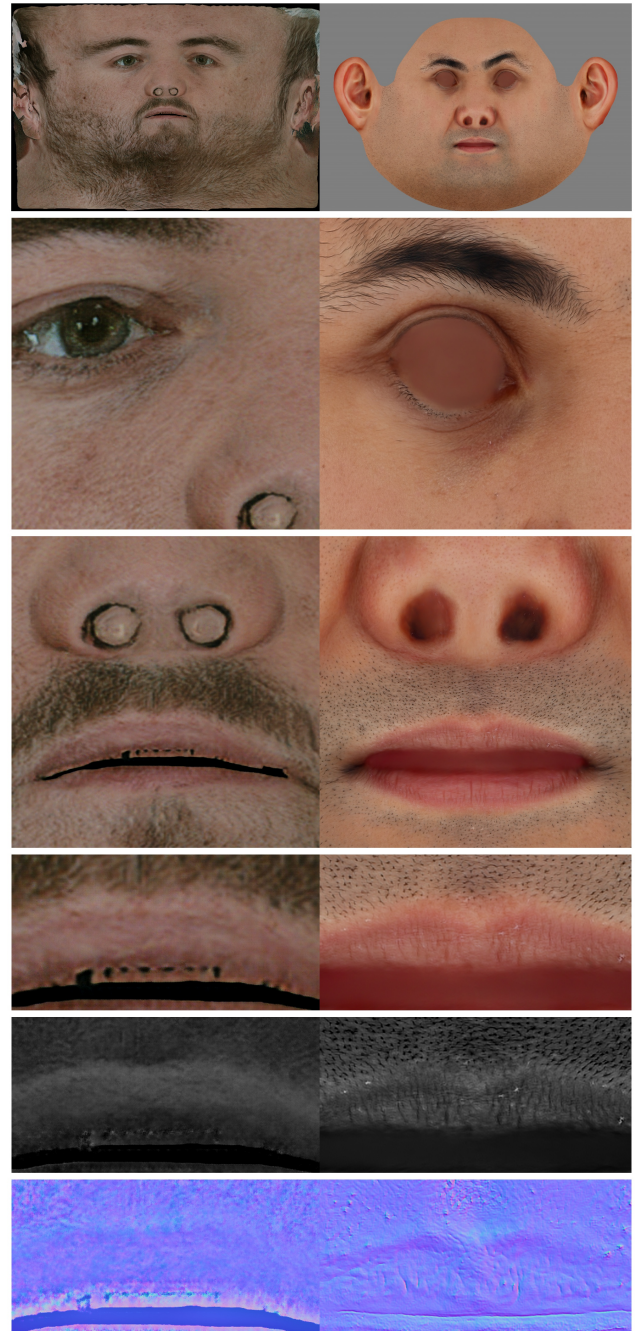


Figure 15: Comparison of *Avatarme++* (left) to our results (right). While both methods generate multi-modal appearance maps of the same resolution, the high frequency detail in our method is much better defined. Please zoom-in to appreciate the differences.

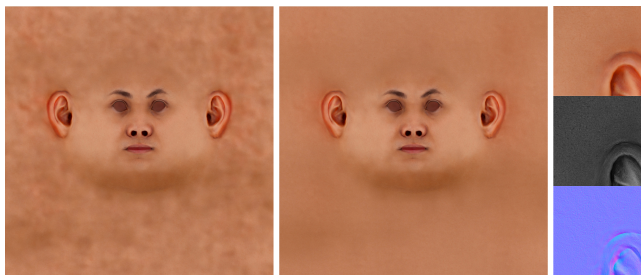


Figure 16: We can extend the synthesis into the untrained regions plausibly by blurring the low frequency content in those regions.

- [LMG*20] LATTAS, ALEXANDROS, MOSCHOGLOU, STYLIANOS, GECER, BARIS, et al. “AvatarMe: Realistically Renderable 3D Facial Reconstruction “in-the-wild””. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, 760–769 [2](#), [8](#).
- [LMP*21] LATTAS, ALEXANDROS, MOSCHOGLOU, STYLIANOS, PLOUMPIIS, STYLIANOS, et al. “AvatarMe++: Facial Shape and BRDF Inference with Photorealistic Rendering-Aware GANs”. *IEEE Transactions on Pattern Analysis & Machine Intelligence* 01 (2021), 1–1 [2](#), [8](#).
- [LMP*23] LATTAS, ALEXANDROS, MOSCHOGLOU, STYLIANOS, PLOUMPIIS, STYLIANOS, et al. *FitMe: Deep Photorealistic 3D Morphable Model Avatars*. 2023. arXiv: [2305.09641 \[cs.CV\]](#) [2](#).
- [LNK*21] LUO, HUIWEN, NAGANO, KOKI, KUNG, HAN-WEI, et al. *Normalized Avatar Synthesis Using StyleGAN and Perceptual Refinement*. 2021. arXiv: [2106.11423 \[cs.CV\]](#) [2](#).
- [LSSS18] LOMBARDI, STEPHEN, SARAGIH, JASON, SIMON, TOMAS, and SHEIKH, YASER. “Deep Appearance Models for Face Rendering”. *arXiv e-prints*, arXiv:1808.00362 (Aug. 2018), arXiv:1808.00362. arXiv: [1808.00362 \[cs.GR\]](#) [2](#).
- [LZWS19] LIU, SHAOHUI, ZHANG, XIAO, WANGNI, JIANQIAO, and SHI, JIANBO. “Normalized diversification”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, 10306–10315 [3](#).
- [MKKY18] MIYATO, TAKERU, KATAOKA, TOSHIKI, KOYAMA, MASANORI, and YOSHIDA, YUICHI. “Spectral Normalization for Generative Adversarial Networks”. *International Conference on Learning Representations*. 2018. URL: <https://openreview.net/forum?id=BlQRgzIT-3>.
- [MLT*19] MAO, QI, LEE, HSIN-YING, TSENG, HUNG-YU, et al. “Mode seeking generative adversarial networks for diverse image synthesis”. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, 1429–1437 [3](#).
- [NSK*22] NTAVELIS, EVANGELOS, SHAHBAZI, MOHAMAD, KASTANIS, IASON, et al. “Arbitrary-scale image synthesis”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022, 11533–11542 [3](#).
- [PKA*09] PAYSAN, PASCAL, KNOTHE, REINHARD, AMBERG, BRIAN, et al. “A 3D Face Model for Pose and Illumination Invariant Face Recognition”. *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*. 2009, 296–301. DOI: [10.1109/AVSS.2009.58](#) [2](#).
- [PLMZ23] PAPANTONIOU, FOIVOS PARAPERAS, LATTAS, ALEXANDROS, MOSCHOGLOU, STYLIANOS, and ZAFEIRIOU, STEFANOS. *Relightify: Relightable 3D Faces from a Single Image via Diffusion Models*. 2023. arXiv: [2305.06077 \[cs.CV\]](#) [2](#).
- [RLNH17] ROTH, KEVIN, LUCCHI, AURELIEN, NOWOZIN, SEBASTIAN, and HOFMANN, THOMAS. “Stabilizing training of generative adversarial networks through regularization”. *Advances in neural information processing systems* 30 (2017) [3](#).
- [SCT*17] SØNDERBY, CASPER KAAE, CABALLERO, JOSE, THEIS, LUCAS, et al. “Amortised MAP Inference for Image Super-resolution”. *International Conference on Learning Representations*. 2017 [3](#).
- [SIE21] SKOROKHODOV, IVAN, IGNATYEV, SAVVA, and ELHOSEINY, MOHAMED. *Adversarial Generation of Continuous Images*. 2021. arXiv: [2011.12026 \[cs.CV\]](#) [2](#).
- [SLNG21] SCHWARZ, KATJA, LIAO, YIYI, NIEMEYER, MICHAEL, and GEIGER, ANDREAS. *GRAF: Generative Radiance Fields for 3D-Aware Image Synthesis*. 2021. arXiv: [2007.02442 \[cs.CV\]](#) [3](#).
- [SLZC22] SHUAI, YANG, LIMING, JIANG, ZIWEI, LIU, and CHANGE, LOY CHEN. “Pastiche Master: Exemplar-Based High-Resolution Portrait Style Transfer”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022 [2](#).
- [SSE21] SKOROKHODOV, IVAN, SOTNIKOV, GRIGORII, and ELHOSEINY, MOHAMED. “Aligning Latent and Image Spaces to Connect the Unconnectable”. *arXiv preprint arXiv:2104.06954* (2021) [2](#).
- [SSK19] SHAMAI, GIL, SLOSSBERG, RON, and KIMMEL, RON. “Synthesizing facial photometries and corresponding geometries using generative adversarial networks”. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 15.3s (2019), 1–24 [2](#).
- [STWW22] SKOROKHODOV, IVAN, TULYAKOV, SERGEY, WANG, YIQUN, and WONKA, PETER. “EpiGRAF: Rethinking training of 3D GANs”. *Advances in Neural Information Processing Systems*. Ed. by OH, ALICE H., AGARWAL, ALEKH, BELGRAVE, DANIELLE, and CHO, KYUNGHYUN. 2022. URL: <https://openreview.net/forum?id=TTM7iEFOTzJ3>.
- [SVR*17] SRIVASTAVA, AKASH, VALKOV, LAZAR, RUSSELL, CHRIS, et al. “Veegan: Reducing mode collapse in gans using implicit variational learning”. *Advances in neural information processing systems* 30 (2017) [3](#).
- [TEB*20a] TEWARI, AYUSH, ELGHARIB, MOHAMED, BHARAJ, GAURAV, et al. “StyleRig: Rigging StyleGAN for 3D Control over Portrait Images, CVPR 2020”. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. June 2020 [2](#).
- [TEB*20b] TEWARI, AYUSH, ELGHARIB, MOHAMED, BR, MALLIKARJUN, et al. “PIE: Portrait Image Embedding for Semantic Control”. Vol. 39. 6. Dec. 2020. DOI: [10.1145/3414685.3417803](#) [2](#).
- [TJL*21] TSENG, HUNG-YU, JIANG, LU, LIU, CE, et al. “Regularizing generative adversarial networks under limited data”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, 7921–7931 [3](#).
- [TTN*21] TRAN, NGOC-TRUNG, TRAN, VIET-HUNG, NGUYEN, NGOC-BAO, et al. “On data augmentation for gan training”. *IEEE Transactions on Image Processing* 30 (2021), 1882–1897 [3](#).
- [VMR*23] VECCHIO, GIUSEPPE, MARTIN, ROSALIE, ROULLIER, ARTHUR, et al. *ControlMat: A Controlled Generative Approach to Material Capture*. 2023. arXiv: [2309.01700 \[cs.CV\]](#) [3](#).
- [WRSJ19] WEBSTER, RYAN, RABIN, JULIEN, SIMON, LOIC, and JURIE, FREDERIC. “Detecting Overfitting of Deep Generative Networks via Latent Recovery”. June 2019, 11265–11274. DOI: [10.1109/CVPR.2019.01153](#) [3](#).
- [XWC*20] XU, RUI, WANG, XINTAO, CHEN, KAI, et al. “Positional Encoding as Spatial Inductive Bias in GANs”. *arxiv*. Dec. 2020 [2](#), [4](#), [5](#).
- [YHJ*19] YANG, DINGDONG, HONG, SEUNGHOO, JANG, YUNSEOK, et al. “Diversity-sensitive conditional generative adversarial networks”. *arXiv preprint arXiv:1901.09024* (2019) [3](#).
- [YOC*22] YOON, DONGHWE, OH, JUNSEOK, CHOI, HAYEONG, et al. *OUR-GAN: One-shot Ultra-high-Resolution Generative Adversarial Networks*. 2022. DOI: [10.48550/ARXIV.2202.13799](#). URL: <https://arxiv.org/abs/2202.13799> [2](#).

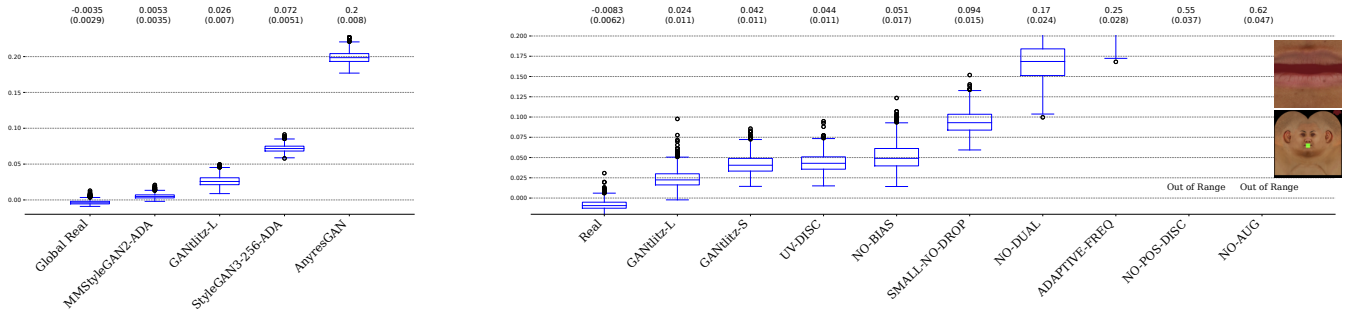


Figure 17: Left: KID value distributions computed on whole albedo texture maps. We compare Data splits of training data against MMStyleGAN2-ADA, GANtLitz-L, StyleGAN3-256-ADA and AnyresGAN. Right: KID values computed on a 512×512 albedo patch of the lips for our models and the different ablations. Section 5.5 introduces the different ablation model identifiers and an explanation on the choice of KID value distributions. The mean (standard deviation) is listed above each box. On the right a depiction of the patch location can be found.

[YTB*21] YENAMANDRA, TARUN, TEWARI, AYUSH, BERNARD, FLORIAN, et al. “i3DMM: Deep Implicit 3D Morphable Model of Human Heads”. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021, 12803–12813 2.

[ZHD*22] ZHOU, XILONG, HAŠAN, MILOŠ, DESCHAINTE, VALENTIN, et al. *TileGen: Tileable, Controllable Material Generation and Capture*. 2022. arXiv: 2206.05649 [cs.GR] 3.

[ZK18] ZHOU, BRADY and KRÄHENBÜHL, PHILIPP. “Don’t let your discriminator be fooled”. *International conference on learning representations*. 2018 3.

[ZLL*20] ZHAO, SHENGYU, LIU, ZHIJIAN, LIN, JI, et al. “Differentiable augmentation for data-efficient gan training”. *Advances in Neural Information Processing Systems* 33 (2020), 7559–7570 3.

[ZQL*23] ZHANG, LONGWEN, QIU, QIWEI, LIN, HONGYANG, et al. *DreamFace: Progressive Generation of Animatable 3D Faces under Text Guidance*. 2023. arXiv: 2304.03117 [cs.GR] 2.

[ZSL*21] ZHAO, ZHENGLI, SINGH, SAMEER, LEE, HONGLAK, et al. “Improved consistency regularization for gans”. *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 12. 2021, 11033–11041 3.