CONFIG: Controllable Neural Face Image Generation

Marek Kowalski, Stephan J. Garbin, Virginia Estellers, Tadas Baltrušaitis, Matthew Johnson, and Jamie Shotton

Microsoft

Abstract. Our ability to sample realistic natural images, particularly faces, has advanced by leaps and bounds in recent years, yet our ability to exert fine-tuned control over the generative process has lagged behind. If this new technology is to find practical uses, we need to achieve a level of control over generative networks which, without sacrificing realism, is on par with that seen in computer graphics and character animation. To this end we propose ConfigNet, a neural face model that allows for controlling individual aspects of output images in semantically meaningful ways and that is a significant step on the path towards finely-controllable neural rendering. ConfigNet is trained on real face images as well as synthetic face renders. Our novel method uses synthetic data to factorize the latent space into elements that correspond to the inputs of a traditional rendering pipeline, separating aspects such as head pose, facial expression, hair style, illumination, and many others which are very hard to annotate in real data. The real images, which are presented to the network without labels, extend the variety of the generated images and encourage realism. Finally, we propose an evaluation criterion using an attribute detection network combined with a user study and demonstrate state-of-the-art individual control over attributes in the output images.

Keywords: neural rendering; face image manipulation; GAN;

1 Introduction

Recent advances in generative adversarial networks (GANs) [17, 18, 5] have enabled the production of realistic high resolution images of smooth organic objects such as faces. Generating photorealistic human bodies, and faces in particular, with traditional rendering pipelines is notoriously difficult [26], requiring handcrafted 3D assets. However, once these assets have been generated we can render the face from any direction and in any pose. In contrast, GANs can be used to easily generate realistic head and face images without the need to author expensive 3D assets, by training on curated datasets of 2D images of real human faces. However, it is difficult to enable meaningful control over this generation without detailed hand labelling of the dataset. Even when conditional models are trained

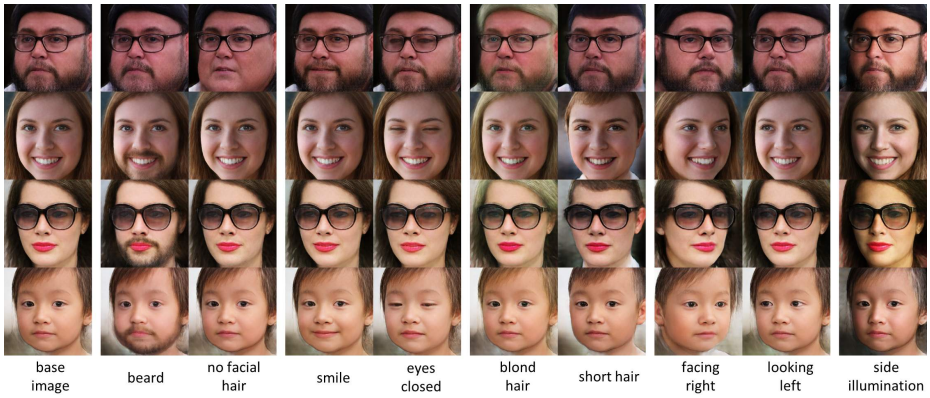


Fig. 1: ConfigNet learns a factorized latent space, where each part corresponds to a different facial attribute. The first column shows images produced by ConfigNet for certain points in the latent space. The remaining columns show changes to various parts of the latent space vectors, where we can generate attribute combinations outside the distribution of the training set like children or women with facial hair.

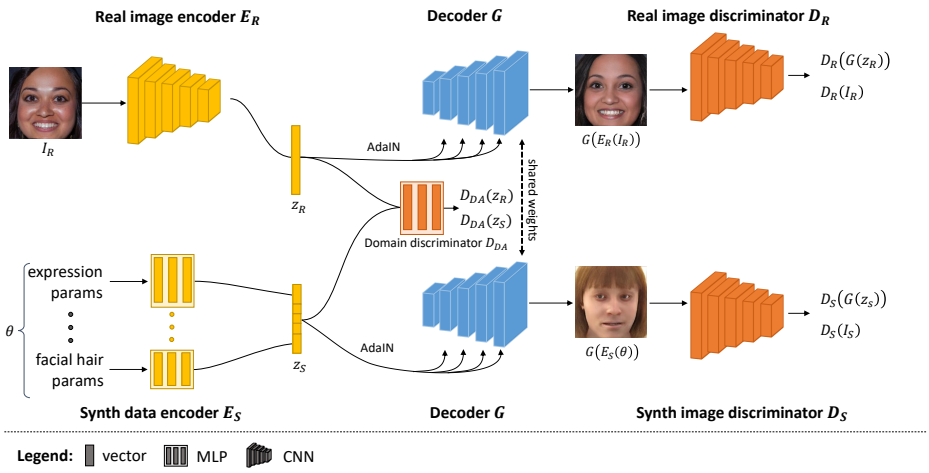


Fig. 2: ConfigNet has two encoders E_R and E_S that encode real face images I_R and the parameters θ of synthetic face images I_S. The encoders output latent space vectors z_R, z_S. The shared decoder, G, generates both real and synthetic images. A domain discriminator D_{DA} ensures the latent distributions generated by E_R and E_S are similar.

with detailed labels, they struggle to generalize to out-of-distribution combinations of control parameters such as children with extensive facial hair or young people with gray hair. In order for GAN based rendering techniques to replace traditional rendering pipelines they must enable a greater level of control.

In this paper we present ConfigNet, one of the first methods to enable control of GAN outputs using the same methods as traditional graphics pipelines. The key idea behind ConfigNet is to train the generative model on both real and synthetically generated face images. Since the synthetic images were generated with a traditional graphics pipeline, the renderer parameters for those images are readily available. We use those known correspondences to train a generative model that uses the same input parametrization as the graphics pipeline used to generate the synthetic data. This allows for independent control of various face aspects including: head pose, hair style, facial hair style, expression and illumination. By simultaneously training the model on unlabelled face images, it learns to generate photorealistic looking faces, while enabling full control over these outputs. Figure 1 shows example results produced by ConfigNet.

ConfigNet can be used to both sample novel images and to embed existing ones, which can then be manipulated. The ability to embed face images sets ConfigNet apart from traditional graphics pipelines, which would require person-specific 3D assets to achieve similar results. The use of a parametrization derived from a traditional graphics pipeline makes ConfigNet easy to use for people familiar with digital character animation. For example, facial expressions are controlled with blendshapes with values in (0, 1), head pose is controlled with Euler angles and illumination can be set using an environment map.

Our main contributions are:

1. ConfigNet, a novel method for placing real and synthetic data into a single factorized and disentangled latent space that is parametrized based on a computer graphics pipeline.
2. A method for using ConfigNet to modify existing face images in a fine-grained way that allows for changing parts of the latent space factors meaningfully.
3. Experiments showing our method generating realistic face images with attribute combinations that are not present in the real images of the training set. For example, a face of a child with extensive facial hair.

2 Related work

Image generation driven by 3D models One of the uses of synthetic data in image generation is the "synthetic to real" scenario, where the goal is to generate realistic images that belong to a target domain based on synthetic images, effectively increasing their realism. The methods that tackle this problem [35, 8, 45] usually use a neural network with an adversarial and semantic loss to push a synthetic image closer to the real domain. While those type of methods can generate realistic images that are controllable through synthetic data, the editing of existing images is difficult as it would require fitting the underlying 3D model to an existing image.

3D model parameters can also be a supervision signal for generative models as shown in [34, 20] and most recently StyleRig [37]. This group of methods shares the challenges of the synthetic to real scenario as they require fitting the 3D model to existing images.

PuppetGAN [41], is a method designed to edit existing images using synthetic data of the same class of objects. It uses two encoder-decoder pairs, one for real and one for synthetic images, which have a common latent space, part of which is designated for an attribute of interest that can be controlled. An image can be edited by encoding it with the real-data encoder, then swapping the attribute of interest part of the latent space with one encoded from a synthetic image and finally decoding with the real-data decoder. Due to the use of separate decoders for real and synthetic data PuppetGAN struggles to decode images where the attribute of interest is outside of the range seen in real data. The method performs well for a single attribute. In contrast, ConfigNet demonstrates disentanglement of multiple face attributes as well as generation of attribute combinations that do not exist in the real training data.

Disentangled representation learning Supervised disentanglement methods try to learn a factorised representation, parts of which correspond to some semantically meaningful aspects of the generated images, based on labelled data in the target domain (as opposed to the synthetic data domain). The major limitation of these methods [24, 44, 29, 7] is that they are only able to disentangle factors of variations that are labelled in the training set. For human faces, labels are easily obtainable for some attributes, such as identity, but the task becomes more difficult with attributes like illumination and almost impossible with attributes like hair style. This labelling problem also becomes more difficult as the required fidelity of the labels increases (e.g. smile intensity).

Unsupervised disentanglement methods share the above goal but do not require labelled data. Most methods in this family, such as β-VAE [12], InfoGAN [6], ID-GAN [21], place constraints on the latent space that lead to disentanglement. The fundamental problem with those approaches is that there is little control over what factors get disentangled and which part of the latent space corresponds to a given factor of variation. HoloGAN [27] separates the 3D rotation of the object in the image from variation in its shape and appearance. ConfigNet borrows the generator architecture of HoloGAN, while disentangling many additional factors of variation and allowing existing images to be edited.

Detach and Adapt [22] is trained in a semi-supervised way on images from two related domains, only one of which has labels. The resulting model allows generating images in both domains with some control over the labelled attribute.

Face video re-enactment Face video re-enactment methods aim to produce a video of a certain person's articulated face that is driven by a second video of the same or a different person. The methods in [38, 44, 42] achieve some of the most impressive face manipulation results seen to date. Face2Face [38] fits a 3D face model and illumination parameters to a video of a person and then re-renders the sequence with modified expression parameters that are obtained from a different sequence. This approach potentially allows for modifying any

aspects of the rendered face that can be modelled, rendered and fitted to the input video. In practice, due to limitations of existing 3D face models and fitting methods, this approach cannot modify complex face attributes like hair style or attributes that require modelling of the whole head, like head pose.

Zakharov et al. [44] propose a video re-enactment method where the images are generated by a neural network driven by face landmarks from a different video sequence. The method produces impressive results given only a small number of target frames. X2Face [42] uses one neural network to resample the source image into a standard reference frame and a second network that resamples this standardized image into a different head pose or facial expression, which can be driven by images or audio signal. While these two methods produce convincing results, the controllability is limited to head pose and expression.

3 Method

The key concept behind the proposed method is to factorize the latent space into parts that correspond to separate and clearly-defined aspects of face images. These factors can be individually swapped or modified (Section 3.6) to modify the corresponding aspect of the output image. The factorization needs labels that fully explain the image content, which would require laborious annotation for real data, but are easily obtained for synthetic data. We thus propose a generative model trained in a semi-supervised way, with labels that are known for synthetic data only. Figure 2 outlines the proposed architecture.

Overview Our approach is to treat the synthetic images $\mathcal{I}_S$ and real images $\mathcal{I}_R$ as two different subsets of a larger set of all possible face images. Hence, the proposed method consists of a decoder G and two encoders E_R and E_S that embed real and synthetic data into a common factorized latent space z (Section 3.1). We will refer to z predicted by E_R and E_S as z_R and z_S respectively. The real data is supplied to its encoder as images $I_R \in \mathcal{I}_R$, while the synthetic data is supplied as vectors $\theta \in \mathbb{R}^m$ that fully describe the content of the corresponding image $I_S \in \mathcal{I}_S$. To increase the realism of the generated images we employ two discriminator networks D_R and D_S for real and synthetic data respectively.

We assume that the synthetic data is a reasonable approximation of the real data so that $\mathcal{I}_S \cap \mathcal{I}_R \neq \emptyset$. Hence, it is desirable for $E_S(\Theta)$ and $E_R(\mathcal{I}_R)$, where Θ is the space of all θ, to also be overlapping. To do so, we introduce a domain discriminator network D_{DA} and train it with a domain adversarial loss [39] on z, that forces z_R and z_S to be close. In Section 4.2 we show that this loss is crucial for the method's ability to control the attributes of the output images.

To accurately reproduce and modify existing images we employ one-shot learning (Section 3.4) that improves reconstruction accuracy compared to embedding using E_R. To enable the sampling of novel images we train a latent GAN that generates samples of z (Section 3.5). Finally, we propose a method for modifying attributes of existing images in a fine-grained way that allows for changing parts of individual factors of z meaningfully (Section 3.6).

3.1 Factorized latent space

Each synthetic data sample θ is factorised into k parts θ_1 to θ_k, such that:

$$\theta \in \mathbb{R}^m = \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \times \ldots \times \mathbb{R}^{m_k}. \tag{1}$$

Each θ_i corresponds to semantically meaningful input of the graphics pipeline used to generate $\mathcal{I}_S$. Examples of such inputs are: facial expression, facial hair parameters, head shape, environment map, etc. The synthetic data encoder E_S maps each θ_i to z_i, a part of z, which thus factorizes z into k parts.

The factorized latent space is a key feature of ConfigNet that allows for easy modification of various aspects of the generated images. For example, one might encode a real image into z using E_R and then change the illumination by swapping out the part of z that corresponds to illumination. Note that the part of z that is swapped in might come from θ_i (encoded by E_S), which is semantically meaningful, or it may come from a different real face image encoded by E_R.

3.2 Loss functions

To ensure that the output image $G(z)$ is close to the ground truth image I_{GT}, we use the perceptual loss $\mathcal{L}_{perc}$ [15], which is the MSE between the activations of a pre-trained neural network computed on $G(z)$ and I_{GT}. We use VGG-19 [36] trained on ImageNet [30] as the pre-trained network. We experimented with using VGGFace [28] as base for the perceptual loss, but didn't see improvement.

While the perceptual loss retains the overall content of the image well, it struggles to preserve some small scale features. Because of that, we use an additional loss with the goal of preserving the eye gaze direction:

$$\mathcal{L}_{eye} = w_M \sum M \circ (I_{GT} - G(z_s)) \quad \text{with} \quad w_M = (1 + |M|_1)^{-1}, \tag{2}$$

where M is a pixel-wise binary mask that denotes the iris, only available for $\mathcal{I}_S$. Thanks to the accurate ground truth segmentation that comes with the synthetic data, similar losses could be added for any part of the face if necessary.

We train the adversarial blocks with the non-saturating GAN loss [9]:

$$\mathcal{L}_{GAN_D}(D, x, y) = \log D(x) + \log(1 - D(y)), \tag{3}$$

$$\mathcal{L}_{GAN_G}(D, y) = \log(D(y)), \tag{4}$$

where $\mathcal{L}_{GAN_D}$ is used for the discriminator and $\mathcal{L}_{GAN_G}$ is used for the generator, D is the discriminator, x is a real sample and y is a network output.

3.3 Two-stage Training procedure

First stage: we train all the sub-networks except E_R, sampling $z_R \sim \mathcal{N}(0, \boldsymbol{I})$ as there is no encoder for real data at this stage. At this stage E_S and G are trained with the following loss:

$$\begin{aligned}\mathcal{L}_1 = {} & \mathcal{L}_{GAN_G}(D_R, G(z_R)) + \mathcal{L}_{GAN_G}(D_{DA}, z_S) \\ & + \mathcal{L}_{GAN_G}(D_S, G(z_S)) + \lambda_{eye}\mathcal{L}_{eye} + \lambda_{perc}\mathcal{L}_{perc}\left(G(z_S), I_S\right),\end{aligned} \tag{5}$$

where $z_S = E_S(\theta)$ and λ are the loss weights. The domain discriminator D_{DA} acts on E_S to bring the distribution of its outputs closer to $\mathcal{N}(0, \boldsymbol{I})$ and so E_S effectively maps the distribution of each θ_i to $\mathcal{N}(0, \boldsymbol{I})$.

Second stage: we add the real data encoder E_R so that $z_R = E_R(I_R)$. The loss used for training E_S and G is then:

$$\mathcal{L}_2 = \mathcal{L}_1 + \lambda_{perc}\mathcal{L}_{perc}\left(G(z_R), I_R\right) + \log(1 - D_{DA}(z_R)), \tag{6}$$

where the goal of $\log(1 - D_{DA}(z_R))$ is to bring the output distribution of E_R closer to that of E_S. In the second stage we increase the weight of λ_{perc}, in the first stage it is set to a lower value as otherwise the total loss for synthetic data would overpower that for real data. Our experiments show that this two-stage training improves controllability and image quality.

3.4 One-shot learning by fine-tuning

Our architecture allows for embedding face images into z using the real data encoder E_R, individual factors z_i can then be modified to modify the corresponding output image as explained in Sections 3.1 and 3.6. We have found that while $G(E_R(I_R))$ is usually similar to I_R as a whole image, there is often an identity gap between the face in I_R and in the generated image. A similar finding was made in [44], where the authors proposed to decrease the identity gap by fine-tuning the generator on the images of a given person.

Similarly, we fine-tune our generator on I_R by minimizing the following loss:

$$\begin{aligned}\mathcal{L}_{ft} = &\mathcal{L}_{GAN_G}(D_R, G(\hat{z_R})) + \log(1 - D_{DA}(\hat{z_R})) \\ &+ \lambda_{perc}[\mathcal{L}_{perc}\left(G(\hat{z_R}), I_R\right) + \mathcal{L}_{face}\left(G(\hat{z_R}), I_R\right)],\end{aligned} \tag{7}$$

where $\mathcal{L}_{face}$ is a perceptual loss with VGGFace [28] as the pre-trained network. We optimize over the weights of G as well as $\hat{z_R}$ which is initialized with $E_R(I_R)$. The addition of a $\mathcal{L}_{face}$ improves the perceptual quality of the generated face images. We believe that this improvement is visible here, but not in the main training phase, as fine-tuning lacks the regularization provided by training on a large number of images and can easily "fool" the single perceptual loss.

3.5 Sampling of z

While the proposed method allows for embedding existing face images into the latent space, sometimes it might be desirable to sample the latent space itself. Samples of the latent space can be used to generate novel images or to sample individual factor z_i. The sampled z_i can then be used to generate additional variations of an existing image that was embedded in z.

To do this, we use a latent GAN [1]. The latent GAN is trained to map between its input $w \sim \mathcal{N}(0, \boldsymbol{I})$ and the latent space z. This simple approach allows for sampling the latent space without the constraints on z imposed by VAEs that lead to reduced quality. The latent GAN is trained with the GAN losses described above, both the discriminator and generator G_{lat} are 3-layer MLPs. Figure 18 in suppl. shows an outline of the method when used with G_{lat}.

3.6 Fine-grained control

Given an existing face image embedded into z, we can easily swap any part, z_i, of its embedding with one that is obtained from E_S or E_R. However, sometimes we might want a finer level of control and only modify a single aspect of z_i while leaving the rest the same. If z_i is a face expression, its single aspect might be the intensity of smile, if z_i is illumination, the brightness might be one aspect. These aspects are controlled by individual elements of the corresponding θ_i vector. However θ_i is unknown if z was generated by E_R or G_{lat}.

For this reason, we use an approximation $\tilde{\theta}_i$ obtained by solving the minimization problem $\min_{\tilde{\theta}_i} |z_i - E_{S_i}(\tilde{\theta}_i)|^2$ with gradient descent, where E_{S_i} is the part of E_S that corresponds to θ_i. We incorporate constraints on θ_i into the optimization algorithm. For example, our expression parameters lie in the convex set $[0, 1]$ and we use projected gradient descent to incorporate the constraint into the minimization algorithm. Given $\tilde{\theta}_i$, e.g. a face expression vector, we can modify the part of the vector responsible for an individual expression and use E_S to obtain a new latent code z_i that generates images where only this individual expression is modified. We use this approach to manipulate individual expressions in Figures 1, 7 and combinations in Figure 12 (supplementary). The method described above is also outlined in Algorithm 1 in supplementary.

3.7 Implementation

The architecture of the decoder G is based on the generator used in HoloGAN [27], explained in supplementary. We choose this particular architecture as it decouples object rotation from z and it allows for specifying the rotation with any parametrization. This lets us obtain the poses of the heads in $\mathcal{I}_S$ in ConfigNet parametrization and supply head pose directly, without an encoder.

The remaining $k - 1$ parts of θ are encoded with separate multi layer perceptrons (MLPs) E_{S_i}, each of which consists of 2 layers with number of hidden units equal to the dimensionality of the corresponding θ_i. The real image encoder E_R is a ResNet-50 [10] pre-trained on ImageNet [31]. The domain discriminator D_{DA} is a 4-layer MLP. The two image discriminators D_R and D_S share the same basic convolutional architecture. The supplementary material contains all network details, source code is available at `http://aka.ms/confignet`.

4 Experiments

Datasets We use the FFHQ [17] (60k images, 1Mpix each), and SynthFace (30k images, 1Mpix each) datasets as a source of real and synthetic training images. We align the face images from all datasets to a standard reference frame using landmarks from OpenFace [4, 43, 2] and reduce the resolution to 256x256 pixels.

Our experiments use the 10k images in the validation set of FFHQ to evaluate ConfigNet. The SynthFace dataset was generated using the method of [3] and setting rotation limits for yaw and pitch to $\pm 30°$ and $\pm 10°$ to cover the typical

Fig. 3: Images from SynthFace dataset, note the domain gap to real images.

Fig. 4: Left: $G(z)$ trained using the two-stage method, where z is sampled from latent GAN. Right: $G(z)$ trained using the first stage only, where z is sampled from prior. Note the large improvement in quality when second stage is added.

range of poses in face images. For SynthFace, θ has $m = 304$ dimensions, while z has $n = 145$ dimensions, and is divided into $k = 12$ factors. Table 6in the supplementary provides the dimensionality of each factor in θ and z, Figure 3 shows sample SynthFace images.

4.1 Evaluation of ConfigNet

Our experiments evaluate ConfigNet key features: photorealism and control.

Photorealism Figure 4 shows samples generated by the latent GAN (where E_R, G were trained using the two stage-procedure of Section 3.3) and a standard GAN model trained only with the first-stage procedure. We observe a large improvement in photorealism when the second stage of training is added. We believe that the low-quality images produced by the standard GAN are caused by the constraint $z \sim \mathcal{N}(0, \boldsymbol{I})$, which is relaxed in our second-stage training thus allowing real and synthetic data to co-exist in the same space.

We quantitatively measure the photorealism and coverage of the generated images using the Frechet Inception Distance (FID) [11] in Table 1. The latent GAN achieves scores that are close to those produced by sampling z through E_R, which is the upper limit of its performance. Training only the first stage and sampling $z \sim \mathcal{N}(0, \boldsymbol{I})$ results in poorer metrics. As expected, the raw synthetic images give the worst result. To further evaluate how much of the photorealism of the generated data is lost due to training on both real and synthetic data, we train ConfigNet without synthetic data and the losses that require its presence. We find that the resulting FID is very close to those produced by our standard training. This suggests that the photorealism of the results might be limited by our network architecture rather then by the use of synthetic data. We speculate

Table 1: FID score for FFHQ, SynthFace, and images obtained with our decoder G and latent codes from the real-image encoder E_R and latent GAN G_{lat}.

Method	FID↓
$G(E_R(I_R))$	33.41
synthetic data $\mathcal{I}_\mathcal{S}$	52.19
$G(E_R(I_R))$ trained without $\mathcal{I}_\mathcal{S}$	33.49
$G(G_{lat}(w)),\ w \sim \mathcal{N}(0,(I))$	39.76
$G(z),\ z \sim \mathcal{N}(0,(I))$ no 2nd stage	43.05

that using a more powerful G and D_R, for example the ones used in StyleGAN, may lead to improved results.

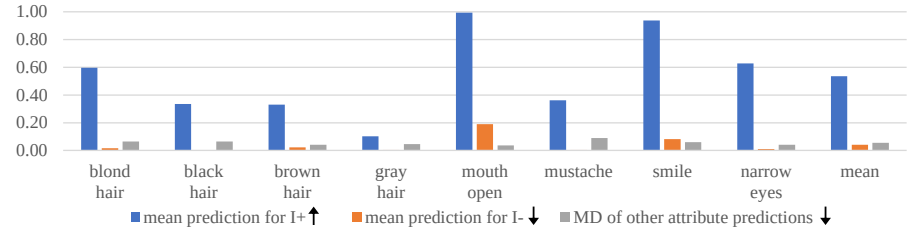
Controllability We evaluate ConfigNet's controllability analysing how changing a specific attribute (e.g., hair colour) changes the output image: with perfect control, the output image should only change with respect to that attribute.

Figure 1 and 6 show controllability qualitatively. Figure 1 shows that the generator is able to modify individual attributes of faces embedded in its latent space, while Figure 6 shows that each attribute can take many different values while only influencing certain aspects of the produced image. The second column of Figure 1 shows that we are able to set facial hair to faces of children and women, demonstrating that the generator is not constrained by the distribution of the real training data. Fine-grained control over individual expressions is shown in Figure 7 as well as Figure 12in the supplementary. The supplementary also includes additional results of face attribute manipulation and interpolation, including a video.

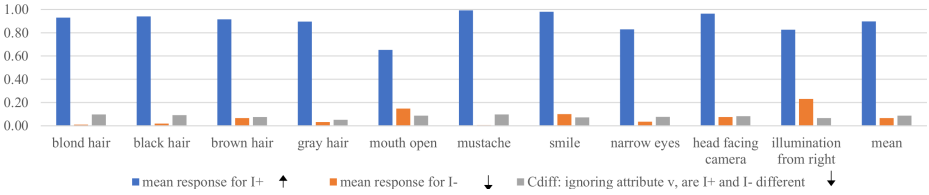
To evaluate if ConfigNet offers this ideal level of control quantitatively, we propose the following experiment: We take a random image I_R from the FFHQ validation set, encode it into latent space $z = E_R(I_R)$ and then swap the latent factor z_i that corresponds to a given attribute v (for example hair colour) with a latent factor obtained with E_S. For each attribute v we output two images: I_+ where the attribute is set to a certain value v_+ (e.g. blond hair) and another I_- with the attribute takes a semantically opposite value v_- (e.g., black hair)[1]. This gives us image pairs (I_+, I_-) that should be identical except for the chosen attribute v, where they should differ. We measure how and where these images differ with an attribute predictor and a user study.

We train an attribute predictor C_{pred} on CelebA [23] to predict 38 face attributes and use it with 1000 FFHQ validation images to estimate 1) if v_+ is present in each set of images pairs (I_+, I_-) and 2) if the other face attributes change. Ideally, $C_{pred}(I_+) = 1,\ C_{pred}(I_-) = 0$ and the Mean Absolute Difference (MD) for other face attributes should be 0. Figure 5a shows how $C_{pred}(I_+) \ggg C_{pred}(I_-)$ while the MD of other attributes is close to 0. The best controllability is achieved for the mouth opening and smile attributes, with $C_{pred}(I_+)$ approaching the ideal value of 1, while the poorest results are achieved for the gray hair attribute. We believe those large differences are caused by bias

[1] We choose the values of Θ_i for v_+ and v_- by manual inspection, details in suppl.



(a) Evaluation of controllability and disentanglement with an attribute predictor.



(b) Evaluation of controllability and disentanglement with a user study.

Fig. 5: Evaluation of control and disentanglement of ConfigNet. Blue and orange bars show the predicted values of given attribute for images with that attribute (I_+, higher better) and images with an opposite attribute (I_-, lower better). The gray bars measure differences of other attributes (MD and C_{diff}, lower better).

in CelebA, where certain attributes are not distributed evenly across age (for example gray hair) or gender (for example moustache).

Our user study C_{user} follows a similar evaluation protocol: 59 users evaluated the presence of v_+ in a total of 1771 images pairs I_+ and I_- on a 5-level scale and gave a score C_{diff} that measures whether, ignoring v, the images depict the same person. Figure 5b shows the results of the controllability and disentanglement metrics for the user study: users evaluate the controllability of the given attribute higher than the feature predictor C_{pred}, with $C_{user}(I_+) > C_{pred}(I_+)$ and $C_{user}(I_+) - C_{user}(I_-) > C_{pred}(I_+) - C_{pred}(I_-)$ for all features except *mouth open*, while the score C_{diff} measuring whether I_+ and I_- show the same person has low values indicating that features other than v_+ remain close to constant. This results support the result of the feature predictor and show a similar performance for different attributes because user judgements do not suffer from the bias of the attribute predictor trained on CelebA.

4.2 Ablation study

We evaluate the importance of two stage training and the domain discriminator D_{DA} by training the neural network without them. Table 2 shows how each of those procedures contributes to controllability of ConfigNet. Compared to the base method, $C_{pred}(I_+) - C_{pred}(I_-)$ decreases by 60% when the domain adversarial loss is removed and by 42% when the first stage training is removed. Quantitatively, the mean absolute difference of the non-altered attributes, MD, is

Table 2: Average controllability metrics for different variants of ConfigNet. D_{DA} denotes the domain discriminator. Ideally, $C_{pred}(I_+) = 1$, $C_{pred}(I_-) = 0$ and MD should be 0. The mean difference $C_{pred}(I_+) - C_{pred}(I_-)$ gives the dynamic range of a given attribute, the higher it is the more controllable the attribute.

Method	$C_{pred}(I_+) \uparrow$	$C_{pred}(I_-) \downarrow$	MD$\downarrow$	$C_{pred}(I_+) - C_{pred}(I_-) \uparrow$
base method	0.54	0.04	0.06	0.50
with fine-tuning	0.52	0.05	0.05	0.47
without D_{DA}	0.39	0.19	0.03	0.20
without 1st stage	0.43	0.14	0.04	0.29

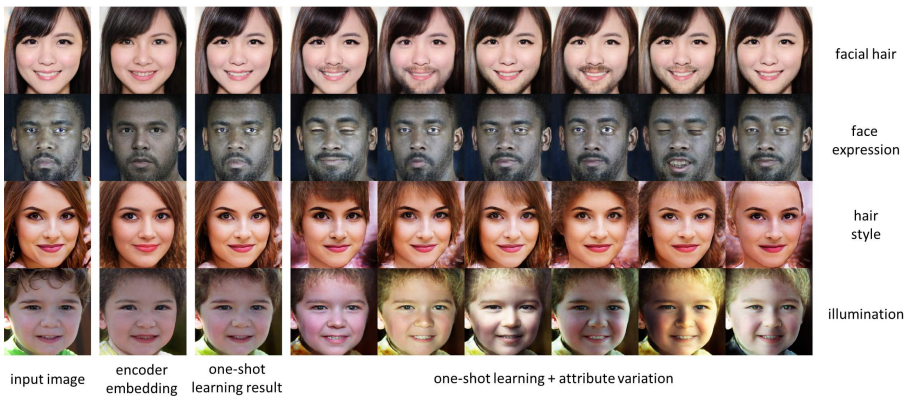


Fig. 6: Effects of fine tuning and attribute variety. The first 3 columns show the input image, the results of the encoder embedding and fine tuning. the other columns show different facial attributes controllable modifying $E_{S_i}(\theta_i)$.

slightly larger for the base method. While this might seem a degradation caused by two stage training and the domain discriminator, we attribute the lower MD to the reduced capability of the network to modify the output image.

One worry with fine-tuning[2] on a single image is that it will change the decoder in a way that negatively affects controllability of the output image. Our experiments show that fine tuning leads to a 6% reduction in $C_{pred}(I_+) - C_{pred}(I_-)$ and no increase of MD, which leads us to believe that the controllability of the fine-tuned generator is not significantly affected. Figure 6 qualitatively shows the effects of fine-tuning compared to embedding using E_R.

An additional ablation study showing the influence of the eye gaze preserving loss $\mathcal{L}_{eye}$ is shown in Figure 16in the supplementary.

4.3 Comparison to state of the art

In this section we compare to PuppetGAN [41], which is the most closely related method, additional comparisons to CycleGAN [45] and Face-ID GAN [34] are in

[2] In all fine-tuning experiments we ran the fine-tuning procedure for 50 iterations.

Fig. 7: Comparison between ConfigNet (left) and PuppetGAN (right). Top row shows the input, left column the desired level of mouth opening for each row. To facilitate comparison, ConfigNet results are cropped to match PuppetGAN.

the supplementary. For comparison to PuppetGAN we use a figure from [41] that shows control over the degree of mouth opening on frames from several videos from the 300-VW dataset [32]. To generate the figure, the authors of PuppetGAN trained separate models on each of the videos and then demonstrated the ability to change the degree of mouth opening in the frames of the same video.

To generate similar results we use a model trained on FFHQ and fine-tune it on the input frame using the method described in Section 3.4. We then use the fine-grained control method (Section 3.6) to change only the degree of mouth opening. The results of this comparison are shown in Figure 7. At a certain level of mouth opening PuppetGAN saturates and is not able to open the mouth more widely, ConfigNet does so, while retaining a similar level of quality and disentanglement. Both methods fail to close the mouth fully for some of the input images. We believe that in case of ConfigNet this is an issue with the disentanglement of the synthetic training set itself, we give further details and describe a solution in supplementary materials. It is also worth noting that PuppetGAN uses hundreds of training images of a specific person, while ConfigNet requires only a single frame and it is able to modify many additional attributes.

4.4 Failure modes

One of the key issues we have identified is that the z_i that corresponds to head shape is often separated for real and synthetic data. For example, changing the head shape of a real image embedded into z using $E_S(\theta_i)$ results in the face appearing closer to the synthetic image space and some of its features being

Fig. 8: Failure modes. Left image pair: changing head shape to one obtained from θ moves the appearance of the image closer to synthetic data. Central image pair: change of z_i corresponding to texture changes style of glasses. Right: frontal image generated from an image I_R with pose outside the supported range.

lost, see Figure 8a for an example. This separation is placed in the head shape space very consistently, we believe this is because head shape affects the whole image in a significant way, so it's easy for the generator to "hide" the difference between real and synthetic images there.

Another issue is that SynthFace does not model glasses, which leads to ConfigNet hiding the representation of glasses in unrelated face attributes, most commonly texture, head and eyebrow shape, as shown in Figure 8b. Lastly, we have found that when I_R has a head pose that is out of the rotation range of SynthFace, the encoder E_R hides the rotation in other parts of z, as shown in Figure 8c. We believe this is a result of constraining the rotation output of E_R to the range seen in SynthFace (details in supplementary). Generating a synthetic dataset with a wider rotation range would likely alleviate this issue.

5 Conclusions

We have presented ConfigNet, a novel face image synthesis method that allows for controlling the output images to an unprecedented degree. Crucially, we show the ability to generate realistic face images with attribute combinations that are outside the distribution of the real training set. This unique ability brings neural rendering closer to traditional rendering pipelines in terms of flexibility.

An open question is how to handle aspects of real face images not present in synthetic data. Adding additional variables in the latent space to model these aspects only for real data is an investigation that we leave to future work.

In the short term, we believe that ConfigNet could be used to enrich existing datasets with samples that are outside of their data distribution or be applied to character animation. In the long term, we hope that similar methods will replace traditional rendering pipelines and allow for controllable, realistic and person-specific face rendering.

Acknowdledgments The authors would like to thank Nate Kushman for helpful discussions and suggestions.

References

1. Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L.: Learning representations and generative models for 3d point clouds. arXiv preprint arXiv:1707.02392 (2017)

2. Baltrusaitis, T., Robinson, P., Morency, L.P.: Constrained local neural fields for robust facial landmark detection in the wild. In: Proceedings of the IEEE international conference on computer vision workshops. pp. 354–361 (2013)
3. Baltrusaitis, T., Wood, E., Estellers, V., Hewitt, C., Dziadzio, S., Kowalski, M., Cashman, T., Johnson, M., Shotton, J.: A high fidelity synthetic face framework for computer vision. Tech. Rep. MSR-TR-2020-24, Microsoft (July 2020), https://www.microsoft.com/en-us/research/publication/high-fidelity-face-synthetics/
4. Baltrusaitis, T., Zadeh, A., Lim, Y.C., Morency, L.P.: Openface 2.0: Facial behavior analysis toolkit. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018). pp. 59–66. IEEE (2018)
5. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096 (2018)
6. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in neural information processing systems. pp. 2172–2180 (2016)
7. Choi, Y., Choi, M., Kim, M., Ha, J.W., Kim, S., Choo, J.: Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8789–8797 (2018)
8. Gecer, B., Bhattarai, B., Kittler, J., Kim, T.K.: Semi-supervised adversarial learning to generate photorealistic face images of new identities from 3d morphable model. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 217–234 (2018)
9. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
11. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in neural information processing systems. pp. 6626–6637 (2017)
12. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. Iclr **2**(5), 6 (2017)
13. Huang, G.B., Ramesh, M., Berg, T., Learned-Miller, E.: Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Tech. Rep. 07-49, University of Massachusetts, Amherst (October 2007)
14. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1501–1510 (2017)
15. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016)
16. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)
17. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4401–4410 (2019)

18. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. arXiv preprint arXiv:1912.04958 (2019)
19. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
20. Kulkarni, T.D., Whitney, W.F., Kohli, P., Tenenbaum, J.: Deep convolutional inverse graphics network. In: Advances in neural information processing systems. pp. 2539–2547 (2015)
21. Lee, W., Kim, D., Hong, S., Lee, H.: High-fidelity synthesis with disentangled representation. arXiv preprint arXiv:2001.04296 (2020)
22. Liu, Y.C., Yeh, Y.Y., Fu, T.C., Wang, S.D., Chiu, W.C., Frank Wang, Y.C.: Detach and adapt: Learning cross-domain disentangled deep representation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8867–8876 (2018)
23. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (December 2015)
24. Mathieu, M.F., Zhao, J.J., Zhao, J., Ramesh, A., Sprechmann, P., LeCun, Y.: Disentangling factors of variation in deep representation using adversarial training. In: Advances in neural information processing systems. pp. 5040–5048 (2016)
25. Mescheder, L., Geiger, A., Nowozin, S.: Which training methods for gans do actually converge? arXiv preprint arXiv:1801.04406 (2018)
26. Mori, M., et al.: The uncanny valley. Energy **7**(4), 33–35 (1970)
27. Nguyen-Phuoc, T., Li, C., Theis, L., Richardt, C., Yang, Y.L.: Hologan: Unsupervised learning of 3d representations from natural images. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 7588–7597 (2019)
28. Parkhi, O.M., Vedaldi, A., Zisserman, A.: Deep face recognition (2015)
29. Qian, S., Lin, K.Y., Wu, W., Liu, Y., Wang, Q., Shen, F., Qian, C., He, R.: Make a face: Towards arbitrary high fidelity face manipulation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 10033–10042 (2019)
30. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
31. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
32. Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: 300 faces in-the-wild challenge: The first facial landmark localization challenge. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 397–403 (2013)
33. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4510–4520 (2018)
34. Shen, Y., Luo, P., Yan, J., Wang, X., Tang, X.: Faceid-gan: Learning a symmetry three-player gan for identity-preserving face synthesis. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 821–830 (2018)
35. Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., Webb, R.: Learning from simulated and unsupervised images through adversarial training. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2107–2116 (2017)
36. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

37. Tewari, A., Elgharib, M., Bharaj, G., Bernard, F., Seidel, H.P., Pérez, P., Zollhofer, M., Theobalt, C.: Stylerig: Rigging stylegan for 3d control over portrait images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6142–6151 (2020)
38. Thies, J., Zollhofer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2face: Real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2387–2395 (2016)
39. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7167–7176 (2017)
40. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Instance normalization: The missing ingredient for fast stylization. arXiv preprint arXiv:1607.08022 (2016)
41. Usman, B., Dufour, N., Saenko, K., Bregler, C.: Puppetgan: Cross-domain image manipulation by demonstration. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9450–9458 (2019)
42. Wiles, O., Sophia Koepke, A., Zisserman, A.: X2face: A network for controlling face generation using images, audio, and pose codes. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 670–686 (2018)
43. Zadeh, A., Chong Lim, Y., Baltrusaitis, T., Morency, L.P.: Convolutional experts constrained local model for 3d facial landmark detection. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. pp. 2519–2528 (2017)
44. Zakharov, E., Shysheya, A., Burkov, E., Lempitsky, V.: Few-shot adversarial learning of realistic neural talking head models. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9459–9468 (2019)
45. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)