One-Shot Domain Adaptation For Face Generation

Chao Yang Ser-Nam Lim

Facebook AI

Abstract

In this paper, we propose a framework capable of generating face images that fall into the same distribution as that of a given one-shot example. We leverage a pre-trained StyleGAN model that already learned the generic face distribution. Given the one-shot target, we develop an iterative optimization scheme that rapidly adapts the weights of the model to shift the output's high-level distribution to the target's. To generate images of the same distribution, we introduce a style-mixing technique that transfers the low-level statistics from the target to faces randomly generated with the model. With that, we are able to generate an unlimited number of faces that inherit from the distribution of both generic human faces and the one-shot example. The newly generated faces can serve as augmented training data for other downstream tasks. Such setting is appealing as it requires labeling very few, or even one example, in the target domain, which is often the case of real-world face manipulations that result from a variety of unknown and unique distributions, each with extremely low prevalence. We show the effectiveness of our one-shot approach for detecting face manipulations and compare it with other few-shot domain adaptation methods qualitatively and quantitatively.

1. Introduction

Deep learning has been prevailing in a variety of computer vision tasks, especially in supervised settings such as learning for classification, detection or segmentation [20, 39, 29, 19]. Deep generative models such as Variational AutoEncoder (VAE) [24, 30] and Generative Adversarial Networks (GAN) [16, 37, 45, 2, 1, 53] in particular have gained significant prominence in the field of deep learning due to their ability to generate highly realistic images depicting faces, natural scenes and objects.

Recent advances in deep learning have paved the way for many important applications ranging from super resolution, movie making, game development, cross domain style transfer, face synthesis and aging prediction, image inpainting, photo editing and others. However, the advent of DL has also precipitated the emerging of applications that abuse its power. Technologies such as Face2Face [45], FaceSwap [2], and encoder-decoder DeepFake [1] have resulted in the rise of online impersonations/fabrication of news, threatening even to sway the outcomes of elections.

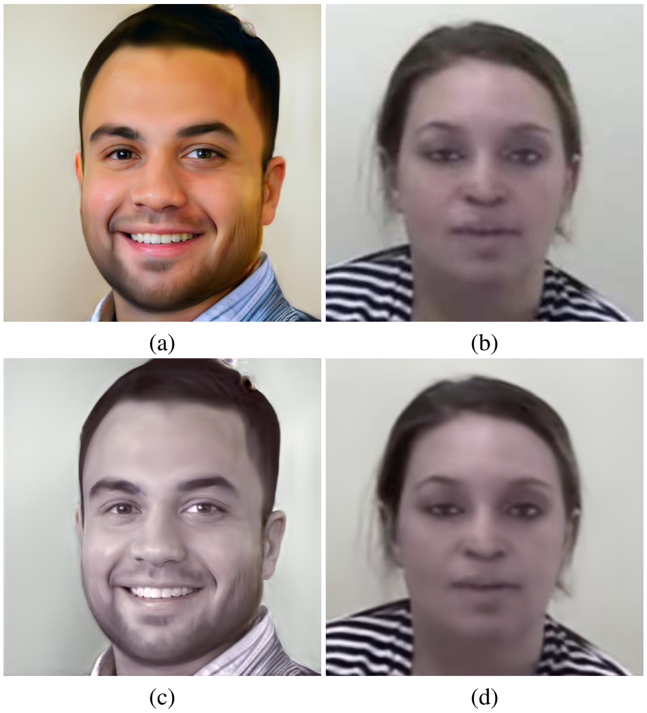


Figure 1: One-shot domain adaptation on encoder-decoder DeepFake using StyleGAN generator. (a). A Random StyleGAN generated image. (b). A one-shot image from encoder-decoder DeepFake of DFDC [13]. (c). A StyleGAN generated image using the same random latent input as (a) after domain adaptation. (d). The StyleGAN reconsutrcted one-shot DeepFake after domain adaptation.

In this paper, we are interested in detecting Deepfakes, which refers to manipulations to replace the human face of authentic media with the face of a different person [40, 10, 13]. This is often coupled with malicious intent of defaming other people or spreading fabricated news. Image generations have been trending as a popular research topic due to the rising prevalence and public interest in Deepfake. As a result, many techniques for generating images have been proposed, each with specific probabilistic distributions or fingerprints. Often times, a new type of synthetic images emerges online but the underlying techniques are unknown, and only a few bespoke examples exist. Training a classifier to detect them poses significant challenges as we are unable to infer the complete probabilistic distribution.

To this end, our method aims to mimic a complete distribution of the target domain given just one example. While most existing domain adaptation approaches try to find a feature space in which there is maximum confusion between source and target distributions [27, 32], we directly manipulate the distribution in the image space. We first train a deep generative model to learn the generic probabilistic distribution of human faces on a large collection of face images. Specifically, we adopt Style-based Generator (StyleGAN) [23] given its high capacity and superior generation quality. With the trained StyleGAN model and a single example from a specific distribution, we run iterative optimization of the input style vector to project the image to the StyleGAN distribution, followed by optimizing the model weights to minimize the projection distance and shift the StyleGAN distribution to the target's. We are then able to generate an unlimited number of random faces that are from a similar distribution as the target and yet preserve the manifold of the generic human face distributions. We further transfer low-level style from the one-shot target to the StyleGAN generated images. To do so, we notice the style vector of a given image has hierarchical structures and represents different attributes at different style layers. If we replace the style vector of randomly synthesized images with the style vector of the target at late layers, we are able to transfer the low-level statistics from the target domain to the random images we generated. We refer to this procedure as style-mixing. Combining iterative weight optimization with style-mixing, the generated images not only capture the overall probabilistic distribution of the target domain but also resemble the one-shot image in low-level appearances and details. Finally, one can then use the generated images to train a model for detecting images from the target domain. Extensive experiment shows that our detector achieves significantly improved accuracy compared with the baseline and other state-of-the-art few-shot domain adaptation and image translation techniques.

Our contributions can be summarized as follows:

1. We introduce a novel one-shot domain adaptation framework that is effective at training a face manipulation detector given a single example of a specific face manipulation distribution.

2. We demonstrate that the images generated with our approach, when utilized to train a classifier, achieves superior performance in telling apart real and manipulated face images.

2. Related Work

2.1. Face Manipulation and Detection

Only recently, a new generation of AI-based image and video synthesis algorithms have become prominent due to the development of new deep generative models such as VAEs [24, 30] and GANs [16, 37]. In this paper, we mostly consider face identity manipulation methods including Face2Face [45], FaceSwap [2], and encoder-decoder Deepfake [1]. Other notable face manipulation methods include audio to lip-sync [44], training a parameter-to-video rendering network [5], synthesizing dynamic-textures with deep neural networks [34], using paired video to learn a conditional GAN [37], or training an identity-specific celebrity network [25]. There are publicly available DeepFake datasets such as FaceForensics++ [40] and DFDC [13]. A comprehensive state-of-the-art report has been published by Zollhofer *et al.* [53].

Our interest in this paper lies in detecting such face manipulations. Existing approaches exploit specific artifacts arising from the synthesis process, such as eye blinking [26], or color, texture and shape cues [11, 9]. Li *et al.* [26] observed that DeepFake faces lack realistic eye blinking, which is utilized in a CNN/RNN model to expose DeepFake videos. Yang *et al.* [49] utilized the inconsistency in head pose to detect fake videos. As a more generic approach, Zhou *et al.* [51] proposed a two-stream CNN for DeepFake detection.

2.2. Few-shot Domain Adaptation

Overcoming the need for large training sets and improving the capability of the model to generalize from few examples have been extensively studied in recent literatures [15, 28, 31, 46]. Earlier work leverages generative models of appearance that share priors across classes in a hierarchical manner [14, 41]. More recently, a new category of works emerges which focuses on using meta-learning to quickly adapt models to novel tasks [15, 35, 38, 33]. These methods adopt better optimization strategies during training and enhance the generalizability of the model. On the other hand, [47, 42, 43] focuses on learning image embeddings that are better suited for few-shot learning. Similarly, [12, 18, 48] also propose augmenting the training set for the few-shot classification task.

2.3. Deep Generative Model for Image Synthesis and Disentanglement

Deep generative models such as GAN [16] and VAE [24] have been very successful in modeling natural image distributions and synthesizing realistic-looking figures. Recent advances such as WGAN [4], BigGAN [7], Progressive GAN [22] and StyleGAN [23] have developed better architectures, losses and training schemes. In particular, StyleGAN [23] proposes a GAN architecture to implicitly learn hierarchical latent styles that contribute to the synthesized images. Our approach leverages StyleGAN as backbone and directly takes advantage of its expressiveness and disentanglement ability. On the other hand, several recent works aim to reverse the generation process and project an image onto latent manifold of GANs, as well as manipulating the latent code to control the output [52, 8, 3, 6]. Our work is motivated to not only manipulate the latent manifold, but also adjust the model-parameter manifold to shift the whole output space given an input image.

3. Our Approach

We first motivate our approach. We are concerned about the scenario where we spot a single face image that is suspected to be generated (aka fake), yet we have no knowledge about the technique that produced it. Our goal is to: (1) Predict the probabilistic distribution of the target given the one-shot example; (2) Sample from the distribution to synthesize random images that are similar to the target domain and; (3) Train a classifier to detect future face images generated by the same technique. At first sight, predicting the distribution of the unknown face manipulation given one example seems ill-posed and unfeasible. We address this by learning a generic face manifold as prior, and then shift the distribution towards the target domain.

3.1. Overview

Our pipeline consists of the following components:

1. **Face Manifold Approximation**. We learn the generic probabilistic distribution of human faces by training StyleGAN on a large collection of natural face images. All possible style vectors of the trained StyleGAN shall span a low-dimensional space that approximates the generic face manifold.

2. **One-shot Manifold Projection**. Given a manipulated face as input, we fix the weight of the StyleGAN model and optimize the style vector to minimize the distance between the synthesized image and the input. Doing so enables us to find the one-shot's nearest neighbor on the StyleGAN manifold. In other words, we *project* the target image onto the manifold.

3. **StyleGAN Manifold Shifting**. After finding the nearest neighbor of the input image, we then fix the corresponding style vector and update the StyleGAN model weights to again minimize the distance between the synthesized image and the target. Updating the weights of the model shifts the output manifold towards the target distribution.

4. **Style Mixing**. We generate a large number of random faces from the updated StyleGAN model. Each time we generate a face, we replace the final layers of the random style vector with those of the target, such that we transfer the low-level statistics from the target to the generated images.

5. **Deepfake Detection**. We use the generated images as training data to learn model for detecting images in the target domain.

3.2. Face Manifold Approximation

Deep generative models are rich, hierarchical models that can learn probability distributions of the training data. As the first step, we resort to these models to learn the generic distribution of faces. We begin by training a deep generative model on a large collections of face images. If the model has sufficient capacity and is well trained, the entirety of its generated images shall span a low-dimension space that approximates the real-world face manifold. Furthermore, given enough training data, the larger capacity the model has, the more closely the output manifold would match with the true face distribution. We consider a few GAN variants including StyleGAN [23], ProGAN [22] and WGAN-GP [17] as candidate models to learn the face manifold.

We analytically examine the capacity and expressiveness of the models by running the following experiment: we first train all three models on real-world face images. After the models are trained, we select one of the models as model A, and fine-tune it with images generated from model B. We then train a classifier on real vs images generated by fine-tuned A, and then test on real vs images generated by B. We can expect that, if A has higher capacity than B, it would learn to generate images with similar distributions and coverages as B. Otherwise, if B is more expressive, it is difficult for fine-tuned A to recover model B's manifold, hence the classification accuracy would be low. Table. 1 lists the classifier generalization results, which clearly shows that StyleGAN is most expressive amongst the candidate models. In addition, StyleGAN generates the most realistic and high-resolution images compared with other generative models. For these reasons, we utilize a StyleGAN model trained on an online collection of high-resolution face images as the base model for our approach.

3.3. StyleGAN Manifold Projection

The original StyleGAN consists of a mapping network f and a synthesis network g. f takes random noise as input and outputs a style vector s. s is modeled as an 18 layer vector. The synthesis network takes the style vector s and a random noise vector as input, and s is used as parameters for adaptive instance normalization [21] to transform the output after each convolution layer. Karras *et*

Model A	Model B	Classification Accuracy
StyleGAN	**ProGAN**	99.6%
	WGAN-GP	99.4%
ProGAN	**StyleGAN**	72.7%
	WGAN-GP	98.1%
WGAN-GP	**StyleGAN**	68.5%
	ProGAN	88.2%

Table 1: Comparing the capacity of StyleGAN, ProGAN and WGAN-GP. Higher classification accuracy indicates A has larger capacity and could better mimic the distribution of B.

al. [22] shows that using style vector as layer-wise guidance not only makes synthesizing high-resolution images easier, but also leads to hierarchical disentanglement of local and global attributes. For our purpose, we consider a trained StyleGAN model. In this case, all possible style vectors generated by the mapping network form a synthetic-face manifold that mimics the true distribution of human faces.

With the StyleGAN manifold and a visual example from an arbitrary distribution, our next step is to project the example onto the manifold. To do so, we first detect the facial landmarks and preprocess the image by cropping it to be 1.3 times larger than the face region, followed by resizing it to 1024x1024 which is the output size of StyleGAN. Let the preprocessed image be I. Projecting I to StyleGAN manifold means we would like to find the style vector s_I that the generated image $g(s_I)$ is most similar to I. In this way, s_I is the style vector corresponding to I's manifold projection. This process could be more formally formulated as solving for the following objective function:

$$s_I = \arg\min_{s} D(g(s), I). \tag{1}$$

With a differentiable distance function D, we can solve for Eqn. 1 by backpropagating the loss D through g with the weights fixed, and then iteratively update s until the loss converges. This is similar to fine-tuning using the given example I, but here we are optimizing s instead of the weights of g. It is also important to use an appropriate distance function D. Common candidates for reconstruction loss are ℓ_1, ℓ_2 and CNN-based perceptual loss. We experimented with those losses and found that using a combination of perceptual and ℓ_1 loss leads to the best visual quality and reconstruction fidelity:

$$D(g(s), I) = \sum_{l} \|f_l(g(s)) - f_l(I)\|_2^2 + \lambda \|g(s) - I\|_1^1. \tag{2}$$

Here f_l is the neuron responses at l_{th} layer extracted with a pre-trained VGG-16 model, and $\lambda = 5$ is the weight of ℓ_1 loss. The reconstruction loss usually converges within 1,000 iterations of optimization. After it converges, the style vector s_I is taken as the projection of I on the StyleGAN manifold, and the reconstruction $g(s_I)$ is the nearest neighbor of I amongst StyleGAN output images.

A more accurate projection requires optimizing the style vector and the noise vector at the same time. However, we found that the noise vector had little effects on the final reconstruction output. In our experiments, we always initialize the style vector to be a zero vector and the noise vector to be random Gaussian, and we update the style vector but keep the noise fixed during optimization.

3.4. StyleGAN Manifold Shifting

After we found the projection of the target on the original StyleGAN manifold, our next step is to shift the StyleGAN manifold towards the target distribution. To do so, we use similar iterative optimization procedure as 3.3. However, instead of updating the style vector s, we fix s to be the output of 3.3 s_I while updating the model weights to match the generated image with the target. The idea here is that every time we update the weight of g, we are slightly adjusting the StyleGAN manifold when the weight changes are sufficiently small. By fixing the style vector to be s_I and updating the model weights of g, we are pulling the nearest neighbor of StyleGAN manifold closer to the target such that the entire manifold becomes more similar to the target distribution. Similar to Eqn. 1, the objective function can be defined as:

$$g_I = \arg\min_{g} D(g(s_I), I). \tag{3}$$

Here, we reuse Eqn. 2 as the distance function. As far as the optimization is concerned, it comes down to the choice of updating different layers of the StyleGAN. As shown in [23], the late layers of the style vector control the low-level details of the output image such as the color or local textures, while the initial layers control the global attributes such as gender, appearance or identity. We experimented updating different StyleGAN layers for manifold shifting, and examine the synthetic image quality and the domain adaptation effectiveness. Our observation is that updating all StyleGAN layers makes the optimized model generate images most similar to the target and also achieve the highest accuracy when being used to train classifiers. Since we already inferred s_I that generates an image similar to I, this step would only slightly adjust the weights of the model. In this case, the optimized model still preserves the generic face manifold learned in 3.2.

To better illustrate the effects of manifold projection and shifting, Fig. 2 shows visual examples of an input, the reconstructed image after manifold projection, and the reconstructed image after manifold shifting. It shows that after manifold shifting, the reconstructed image matches with the input more closely in global color and appearances.

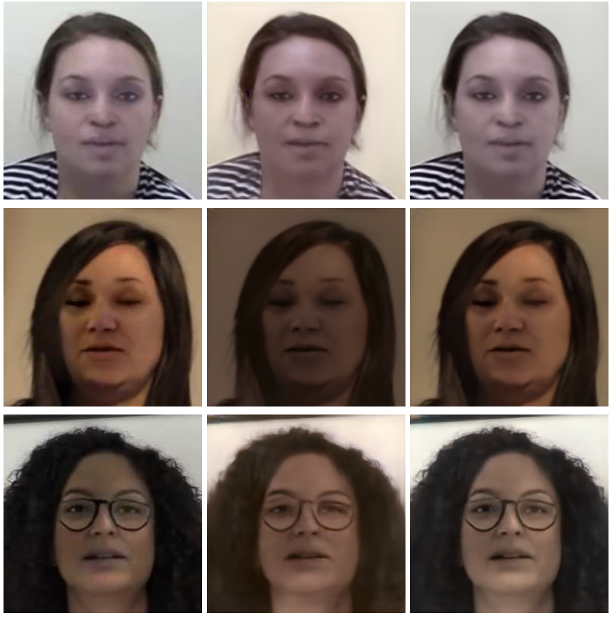


Figure 2: The reconstruction after manifold projection and manifold shifting for one-shot encoder-decoder (top), neural talking head (middle) and FSGAN (bottom). From left to right: the input image; $g(s_I)$ as the reconstructed image after manifold projection; $g_I(s_I)$ as the reconstructed image after manifold shifting.

3.5. Style Mixing

In the previous steps, we optimize the weights of the StyleGAN model such that it generates images of similar distribution as the target. However, it does not suffice to simply change the global appearances as there are certain low-level statistics that the target exhibits, which are useful signals when training a classifier. We propose to use style mixing to further generate images that match with the target in low-level details. Karras *et al.* [23] shows that StyleGAN's style vector comes with the property of disentanglement, which separates the high-level and low-level attributes of the synthesized image at different style layers. Based on this, we use the style vector s_I inferred in 3.3 as the interpretable representation of the target. For each random style vector s that we sampled with the mapping network, we replace the final layers of s with those of s_I before giving it as input to the generator so that the generated random image $g(s)$ inherits the low-level color and textures from I.

We experimented with replacing different number of the final layers, and found that replacing the three last layers of s with those of s_I preserves the global appearance of the image yet still manages to change the output to more closely resemble I. Combining manifold shifting and style mixing, the generated images not only capture the generic human face manifold but also display low-level statistics of the target.

3.6. Classification

The last step is to use the randomly generated images as a synthetic dataset to train a classifier against the target domain. In the case of face manipulation detection, we train a classifier between real images and one-shot optimized StyleGAN synthetic images, and use it to detect the actual face manipulation images from the real images. Another task we could solve using the synthetic datasets is multi-domain classification, where given an image we could classify it into a specific type of face manipulations. For all classification tasks, we use ResNet50 [20] as the backbone model.

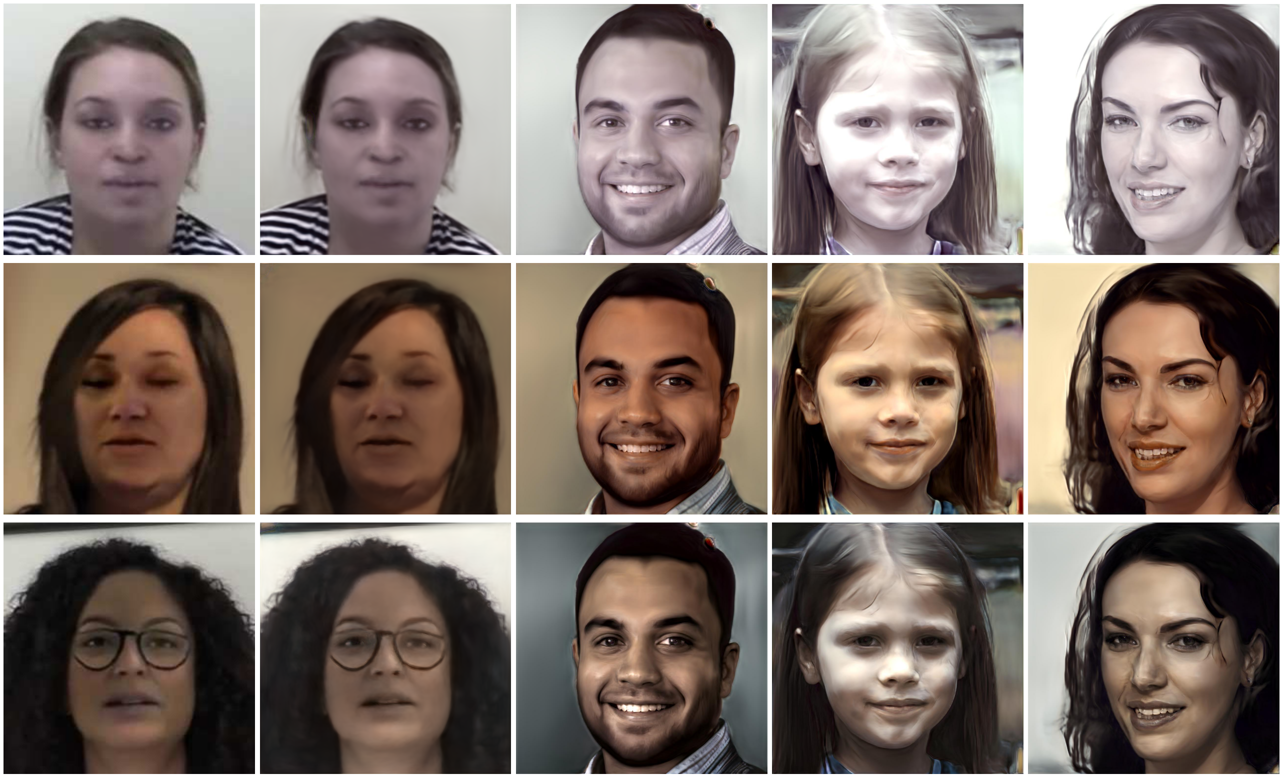
4. Experiments

4.1. Experiment Setup and Results

We evaluate our method on several face manipulation algorithms to show its effectiveness. We use DFDC [13] and FaceForensics++ [40], which consists of a large number of videos that are generated by different face manipulation techniques including encoder-decoder Deepfake [1], Face2Face [45] and FaceSwap [2], neural talking head [50] and FSGAN [36]. For each of the algorithms, we randomly select one image from the dataset. We then apply our one-shot domain adaptation to shift the StyleGAN distribution towards the image and mix the low-level styles and generate a large number of random faces. Finally, we train a classifier using the generated faces to detect images in the target domain.

Qualitatively, we show visual results of one-shot domain adaptation in Fig. 3. For each dataset, we show the one-shot image, the reconstructed image corresponding to the target, and randomly sampled images that mimic the target distribution. At each column, we use the same random style vector to generate the images so that they have the same identities. However, their appearances change according to the one-shot input. We can see that the reconstructed image, which is the closest neighbor of the target on the shifted StyleGAN manifold, visually resembles the target. The randomly sampled images also inherit similar appearances and low-level characteristics from the target.

For quantitative evaluation, we first generate 10,000 random images using the one-shot example for each of the face manipulation techniques. We then train a classifier using real face images as real and the 10,000 synthetic images as fake, and then test on real face images vs the actual face manipulation images (encoder-decoder, Face2Face or FaceSwap). As the baseline, we train the classifier using real faces images and 10,000 random images generated by the original StyleGAN (without domain adaptation) and evaluate on real face images vs actual face manipulations. Table. 2 shows the results. From it we can see that without



(a) Target (b) Reconstructed image (c) Randomly sampled images.

Figure 3: Visual examples of one-shot DeepFake domain adaptation using StyleGAN. From top to bottom: encoder-decoder, neural talking head and FSGAN.

domain adaptation, the detection accuracy is low. This is expected as the original StyleGAN generated images have a distinct distribution from the target. However, after domain adaptation using the one-shot example, the classification accuracy improves significantly - almost perfect for all three datasets. This shows that our one-shot domain adaptation is effective at generating images with similar distribution to the target domain, at the slight cost of seeing one more image compared with the baseline. In addition to train a binary classifier on real vs manipulated, we further experimented with fine-grained classification by training a multi-domain classifier using all images from the three StyleGAN-synthetic face manipulation domains. The high accuracy of fine-grained classification (82.1%) shows that the our synthetic datasets have distinguishable distributions from each other and their distributions are also consistent with the target domain.

In Fig. 4, we plot the t-SNE embeddings of StyleGAN generated image before and after domain adaptation, comparing with the embeddings of actual fake images. We can see that before domain adaptation, the embeddings of StyleGAN generated images and encoder-decoder Deepfake images are separated. After domain adaptation, the embeddings between the two domains are much closer to each other.

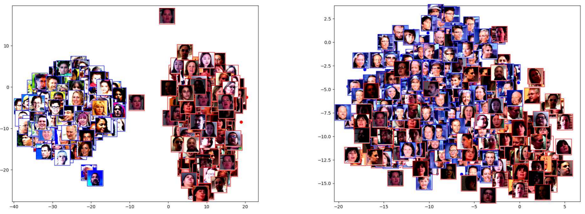


Figure 4: t-SNE embedding visualizations. Left: Embeddings of original StyleGAN generated images and encoder-decoder Deepfake images. Right: Embeddings of one-shot domain-adapted StyleGAN generated images and Deepfake images.

4.2. Ablation Study

As described, there are two main components of our approach: StyleGAN manifold shifting and style mixing. To better understand the effects of each component, in Fig. 5 we compare visual examples of a randomly generated image with manifold shifting only and with style mixing only, given the one-shot example. We can see that with only man-

Train	Test	Average Precision
Real/StyleGAN	Real/encoder-decoder	35.2%
Real/one-shot encoder-decoder	Real/encoder-decoder	93.4%
Real/StyleGAN	Real/Face2Face	35.3%
Real/one-shot Face2Face	Real/Face2Face	99.2 %
Real/StyleGAN	Real/FaceSwap	41.6%
Real/one-shot FaceSwap	Real/FaceSwap	95.2%
Real/(one-shot) encoder-decoder/Face2Face/FaceSwap	Real/encoder-decoder/Face2Face/FaceSwap	82.1%

Table 2: Quantitative evaluation results. (One-shot) encoder-decoder Deepfake/Face2Face/FaceSwap are the synthetic datasets generated by StyleGAN after running domain adaptation algorithm given the one-shot example of encoder-decoder Deepfake/Face2Face/FaceSwap (Fig. 3 (a)).

ifold shifting, the output images are different from the input in low-level characteristics. Meanwhile, if we only mix the styles without adjusting the original StyleGAN model, the output image's colors and textures are distorted and do not look realistic. In both cases, the classification accuracy using the randomly generated images significantly drops compared with the results of utilizing both components (Table. 3).

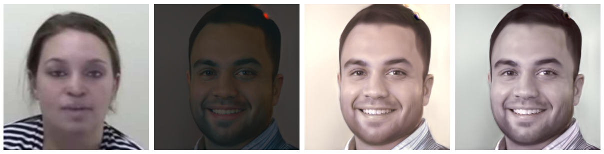


Figure 5: Analyzing the effects of different components. From left to right: Target encoder-decoder Deepfake image; Randomly generated image after manifold shifting without style mixing; Randomly generated image using original StyleGAN with style mixing; Randomly generated image after manifold shifting and style mixing.

Setting	Average Precision
StyleGAN manifold shifting only	43.1%
StyleGAN mix style only	34.0%
Ours	93.4%

Table 3: Real vs encoder-decoder Deepfake classification results. The classification models are trained using images generated by StyleGAN with manifold-shifting only or style-mixing only.

As for the reconstruction losses used for optimizing the style vector and the model weights, we experimented with ℓ_1, ℓ_2, VGG-16, and a combination of these losses. We observe that the reconstruction quality is correlated with the reconstruction loss used. As shown by the example images in Fig. 6, using a single loss of either ℓ_1, ℓ_2 or VGG-16 results in bleached color or distorted appearances compared with accurate reconstructions generated by using a combination of ℓ_1 and perceptual loss. Note that for all VGG-16 losses, we use all the layers to compute the feature responses when measuring the perceptual similarity.

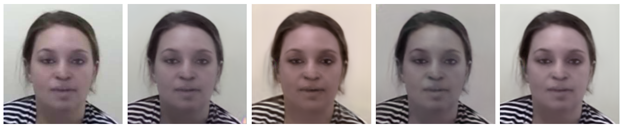
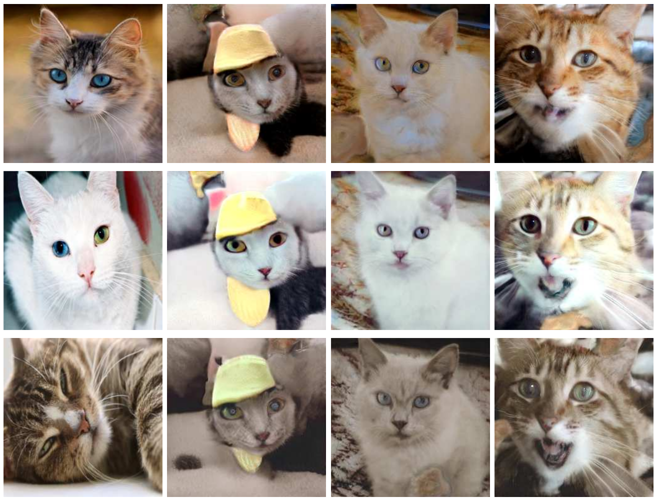


Figure 6: Input reconstruction after optimizing style and weights using different losses. From left to right: input, ℓ_1 loss, ℓ_2 loss, VGG-16 loss, combining ℓ_1 and VGG-16 losses.

To show the effectiveness of our approach on datasets other than face manipulations and that it can be used as a generic domain adaptation approach, we further show that our one-shot domain adaptation technique can be applied to other domains such as cats. Given a single cat image as input and a pre-trained StyleGAN cat model, we can generate random images that are visually similar to the target (Fig. 7).

4.3. Comparisons

Comparison with Few-shot Classification We compare our results with few-shot classifications. We train a classifier using different number of examples from the target domain (encoder-decoder Deepfake) and use it to classify the target from the real faces. Table. 4 shows the results. We can see that directly training a classifier with very few examples (1 or 10) in the target domain leads to inferior performance compared with ours. Only when the number of examples from the target domain is large enough (over 100), the classifier can achieve high accuracy. For all the experiments, we use 10,000 real face images as real, and adjust the weight between false positive loss and false negative loss to reflect the imbalanced quantity of real and fake.
Comparison with Fine-tuning Another possibility is to fine-tune a pre-trained StyleGAN model, by further training it with a few examples from the target domain. Ide-



(a) Input (b) Randomly generated images

Figure 7: One-shot domain adaptation on cats.

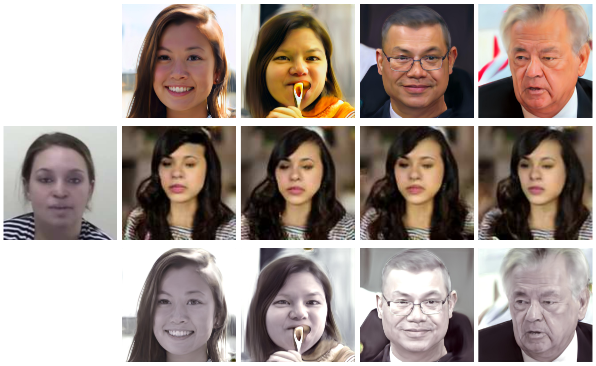
Train	Test	Average Precision
Real/1-shot DF Clf	Real/DF	52.1%
Real/10-shot DF Clf	Real/DF	79.7%
Real/100-shot DF Clf	Real/DF	93.0%
Real/1000-shot DF Clf	Real/DF	99.5%
Ours		93.4

Table 4: Comparing with training a encoder-decoder Deepfake classifier with different number of examples. Our one-shot domain adaptation can achieve detection accuracy on par with a 100-shot DF classifier.

ally the fine-tuned model would generate synthetic images with a similar distribution as the target domain. However, we found that fine-tuning StyleGAN with only a few examples is difficult, as the model would collapse and keep generating identical images. The classification accuracy of Table 5 shows that only when we have sufficient examples (over 100) to fine-tune the original StyleGAN model, would it not lead to mode collapse and achieve reasonable classification accuracy. Note here we use the ProGAN (pre-trained on real face images) synthetic images as the target domain, which is more difficult to differentiate with real.

Train	Test	Average Precision
Real/1-shot ProGAN	Real/ProGAN	10.2%
Real/10-shot ProGAN	Real/ProGAN	21.8%
Real/100-shot ProGAN	Real/ProGAN	88.7%
Real/1000-shot ProGAN	Real/ProGAN	99.0%
Ours		62.1%

Table 5: Comparing with fine-tuning StyleGAN with different number of examples.



Method	FUNIT	Ours
Accuracy	34.4%	93.4%

Figure 8: Above from top to bottom: Random StyleGAN generated images; Input 1-shot encoder-decoder Deepfake and translated images using FUNIT; Translated images using our one-shot domain adaptation. Below: DeepFake classification accuracy when trained on FUNIT vs our generated images.

Comparison with FUNIT We also compare our results with Few-Shot Unsupervised Image-to-Image Translation (FUNIT) [28]. FUNIT could also translate an image to the target domain given a few examples of the target domain. However, at training time, FUNIT would require a large number of labeled images in over 100 classes. In contrast, our approach is entirely unsupervised and only requires pretraining the StyleGAN model. At test time, FUNIT can also translate the images from the source domain (e.g. StyleGAN synthetic images) to the target domain (e.g. encoder-decoder Deepfake) given one example. However in terms of translation quality, from Fig. 8 we can see that FUNIT actually modifies the identity of the source image instead of changing the appearances or low-level statistics. We further use these images to train a real/Deepfake classifier: we first use random StyleGAN generated images as the content and the 1-shot Deepfake image as the style, which are given to the trained FUNIT model to generate a synthetic dataset that adapts StyleGAN to Deepfake. We then train a classifier using real vs FUNIT translated images, and test on real vs actual Deepfake images. This results in an average precision significantly lower than ours.

5. Conclusions

We propose a simple yet effective one-shot domain adaptation method based on StyleGAN. Our approach not only generates compelling visual results similar to the one-shot target, but also allows us to train robust classifiers in response to different target domains. This process is also fully automatic, requiring little supervision. As future work, we would like to extend our framework to be a more generic image translation and domain adaptation approach.

References

[1] Deepfakes github. https://github.com/deepfakes/faceswap. Accessed: 2019-11-05. 1, 2, 5

[2] Faceswap. https://github.com/MarekKowalski/FaceSwap/. Accessed: 2019-11-05. 1, 2, 5

[3] R. Abdal, Y. Qin, and P. Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4432–4441, 2019. 3

[4] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 3

[5] H. Averbuch-Elor, D. Cohen-Or, J. Kopf, and M. F. Cohen. Bringing portraits to life. *ACM Transactions on Graphics (Proceeding of SIGGRAPH Asia 2017)*, 36(6):196, 2017. 2

[6] D. Bau, J.-Y. Zhu, H. Strobelt, B. Zhou, J. B. Tenenbaum, W. T. Freeman, and A. Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. *arXiv preprint arXiv:1811.10597*, 2018. 3

[7] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 3

[8] A. Brock, T. Lim, J. M. Ritchie, and N. Weston. Neural photo editing with introspective adversarial networks. *arXiv preprint arXiv:1609.07093*, 2016. 3

[9] T. Carvalho, F. A. Faria, H. Pedrini, R. d. S. Torres, and A. Rocha. Illuminant-based transformed spaces for image forensics. *IEEE transactions on information forensics and security*, 11(4):720–733, 2015. 2

[10] R. Chesney and D. K. Citron. Deep fakes: a looming challenge for privacy, democracy, and national security. 2018. 1

[11] T. J. De Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. de Rezende Rocha. Exposing digital image forgeries by illumination color classification. *IEEE Transactions on Information Forensics and Security*, 8(7):1182–1194, 2013. 2

[12] M. Dixit, R. Kwitt, M. Niethammer, and N. Vasconcelos. Aga: Attribute-guided augmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7455–7463, 2017. 2

[13] B. Dolhansky, R. Howes, B. Pflaum, N. Baram, and C. C. Ferrer. The deepfake detection challenge (dfdc) preview dataset. *arXiv preprint arXiv:1910.08854*, 2019. 1, 2, 5

[14] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence*, 28(4):594–611, 2006. 2

[15] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017. 2

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 1, 2

[17] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017. 3

[18] B. Hariharan and R. Girshick. Low-shot visual recognition by shrinking and hallucinating features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3018–3027, 2017. 2

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 1

[20] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 5

[21] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 3

[22] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 3, 4

[23] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 2, 3, 4, 5

[24] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1, 2

[25] I. Korshunova, W. Shi, J. Dambre, and L. Theis. Fast face-swap using convolutional neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3677–3685, 2017. 2

[26] Y. Li, M.-C. Chang, and S. Lyu. In ictu oculi: Exposing ai created fake videos by detecting eye blinking. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7. IEEE, 2018. 2

[27] A. H. Liu, Y.-C. Liu, Y.-Y. Yeh, and Y.-C. F. Wang. A unified feature disentangler for multi-domain image translation and manipulation. In *Advances in Neural Information Processing Systems*, pages 2590–2599, 2018. 2

[28] M.-Y. Liu, X. Huang, A. Mallya, T. Karras, T. Aila, J. Lehtinen, and J. Kautz. Few-shot unsupervised image-to-image translation. *arXiv preprint arXiv:1905.01723*, 2019. 2, 8

[29] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 1

[30] S. Lombardi, J. Saragih, T. Simon, and Y. Sheikh. Deep appearance models for face rendering. *ACM Transactions on Graphics (TOG)*, 37(4):68, 2018. 1, 2

[31] S. Motiian, Q. Jones, S. Iranmanesh, and G. Doretto. Few-shot adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 6670–6680, 2017. 2

[32] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5715–5725, 2017. 2

[33] T. Munkhdalai and H. Yu. Meta networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2554–2563. JMLR. org, 2017. 2

[34] K. Nagano, J. Seo, J. Xing, L. Wei, Z. Li, S. Saito, A. Agarwal, J. Fursund, H. Li, R. Roberts, et al. pagan: real-time avatars using dynamic textures. *ACM Trans. Graph.*, 37(6):258–1, 2018. 2

[35] A. Nichol, J. Achiam, and J. Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018. 2

[36] Y. Nirkin, Y. Keller, and T. Hassner. Fsgan: Subject agnostic face swapping and reenactment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7184–7193, 2019. 5

[37] K. Olszewski, Z. Li, C. Yang, Y. Zhou, R. Yu, Z. Huang, S. Xiang, S. Saito, P. Kohli, and H. Li. Realistic dynamic facial textures from a single image using gans. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5429–5438, 2017. 1, 2

[38] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. 2016. 2

[39] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 1

[40] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. Faceforensics++: Learning to detect manipulated facial images. *arXiv preprint arXiv:1901.08971*, 2019. 1, 2, 5

[41] R. Salakhutdinov, J. Tenenbaum, and A. Torralba. One-shot learning with a hierarchical nonparametric bayesian model. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pages 195–206, 2012. 2

[42] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017. 2

[43] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. 2

[44] S. Suwajanakorn, S. M. Seitz, and I. Kemelmacher-Shlizerman. Synthesizing obama: learning lip sync from audio. *ACM Transactions on Graphics (TOG)*, 36(4):95, 2017. 2

[45] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2387–2395, 2016. 1, 2, 5

[46] S. Valverde, M. Salem, M. Cabezas, D. Pareto, J. C. Vilanova, L. Ramió-Torrentà, À. Rovira, J. Salvi, A. Oliver, and X. Lladó. One-shot domain adaptation in multiple sclerosis lesion segmentation using convolutional neural networks. *NeuroImage: Clinical*, 21:101638, 2019. 2

[47] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016. 2

[48] Y.-X. Wang, R. Girshick, M. Hebert, and B. Hariharan. Low-shot learning from imaginary data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7278–7286, 2018. 2

[49] X. Yang, Y. Li, and S. Lyu. Exposing deep fakes using inconsistent head poses. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8261–8265. IEEE, 2019. 2

[50] E. Zakharov, A. Shysheya, E. Burkov, and V. Lempitsky. Few-shot adversarial learning of realistic neural talking head models. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9459–9468, 2019. 5

[51] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis. Two-stream neural networks for tampered face detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1831–1839. IEEE, 2017. 2

[52] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016. 3

[53] M. Zollhöfer, J. Thies, P. Garrido, D. Bradley, T. Beeler, P. Pérez, M. Stamminger, M. Nießner, and C. Theobalt. State of the art on monocular 3d face reconstruction, tracking, and applications. In *Computer Graphics Forum*, volume 37, pages 523–550. Wiley Online Library, 2018. 1, 2