

# DepthwiseGANs: Fast Training Generative Adversarial Networks for Realistic Image Synthesis

Mkhuseli Ngxande

CSIR Defence, Peace, Safety and  
Security

Optronic Sensor Systems,  
Pretoria, South Africa

Email: mngxande@csir.co.za

Jules-Raymond Tapamo

School of Computer Engineering  
University of Kwa-Zulu Natal,

Durban, South Africa

Email: tapamoj@ukzn.ac.za

Michael Burke

Mobile Intelligent Autonomous Systems  
Modelling and Digital Science

Council for Scientific and Industrial Research  
Pretoria, South Africa

Email: michaelburke@ieee.org

**Abstract**—Recent work has shown significant progress in the direction of synthetic data generation using Generative Adversarial Networks (GANs). GANs have been applied in many fields of computer vision including text-to-image conversion, domain transfer, super-resolution, and image-to-video applications. In computer vision, traditional GANs are based on deep convolutional neural networks. However, deep convolutional neural networks can require extensive computational resources because they are based on multiple operations performed by convolutional layers, which can consist of millions of trainable parameters. Training a GAN model can be difficult and it takes a significant amount of time to reach an equilibrium point. In this paper, we investigate the use of depthwise separable convolutions to reduce training time while maintaining data generation performance. Our results show that a DepthwiseGAN architecture can generate realistic images in shorter training periods when compared to a StarGan architecture, but that model capacity still plays a significant role in generative modelling. In addition, we show that depthwise separable convolutions perform best when only applied to the generator. For quality evaluation of generated images, we use the Fréchet Inception Distance (FID), which compares the similarity between the generated image distribution and that of the training dataset.

**Index Terms**—Synthetic Data, GANs, Depthwise Separable Convolution, FID.

## I. INTRODUCTION AND RELATED WORK

Image-to-image translation is used in a wide variety of applications, including style transfer, super-resolution, and face synthesis [1, 2, 3, 4]. Given sufficient training data, these techniques allow one to translate an input image to a desired output, for example, changing facial attributes from a male to a female (see Figure 1). Early work on image-to-image translation by Hertzman et al. used a non-parametric mixture model using a single input-output training image pair [5]. The introduction of Generative Adversarial Networks (GAN) has resulted in great success in synthesizing realistic images [6, 7, 8]. Traditional GAN architectures consist of two deep convolutional neural networks (generator and discriminator), which play a mini-max game against each other. The generator produces synthetic data from a given latent sample, while the discriminator evaluates the generator’s output against true data. Therefore, the generator is forced to produce appealing images to fool the discriminator. However, this competing behaviour makes it difficult to train GANs, which often require

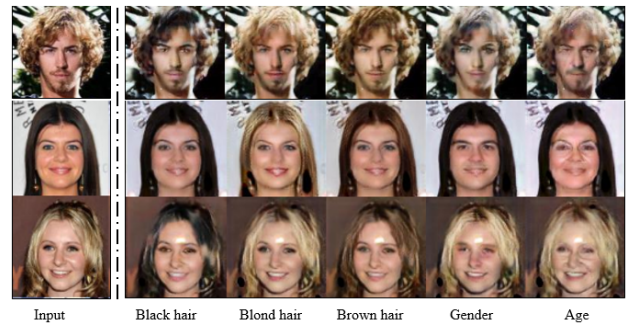


Figure 1. Multi-domain image translation on the CelebA dataset using our proposed model. Given an input image from one domain, we can translate it to another, for example, changing hair color, gender or age.

extensive computational resources [9, 10, 11]. Recent research has focused on ways to stabilize the training process and using lightweight networks to speed up the process of training. The work of Arjovsky et al. proposed minimizing the Earth Mover (EM) distance in the learning distribution which helps the training process [12]. In addition to improving the training process, [13] introduced a method that uses the estimated difference measure from the discriminator to select important weights. This provides a gradient for training the generator network and also improves the stability of the training process. Moving beyond traditional GANs, there is a wide interest in controllable GANs, for example conditional GANs, which introduce additional information to a generator to produce conditioned images [1, 14]. The StarGan architecture provides a scalable approach to perform multi-domain image-to-image translation using a single model. This is accomplished by relying on an auxiliary classifier to group labelled input data into multiple domains. This produces high quality, but relatively low resolution, synthetic images.

In this paper, we investigate the effects and benefits of substituting the traditional convolutional layers in StarGan with depthwise separable convolutions. Depthwise separable convolutions are a special case of group convolution [15] that reduce the number of parameters in a convolution by independently applying smaller convolution filters to input tensor channels, using a 1 dimensional end stage convolution

to group information from separate channels. This reduces the total number of operations required by the model. Group convolutions have already been shown to be an effective means of building lightweight models [16], but this work shows that they can affect model capacity. Our primary findings are as follows:

- Introducing depthwise separable convolutions in the generator reduces the number of parameters, thus resulting in faster network training.
- The inclusion of depthwise separable convolutions in the generator is more effective than when applied in both the generator and discriminator.
- However, the quality of the generated images is largely dependent on network capacity, and additional layers are required to recover from capacity loss resulted from the use of depthwise separable convolutions.

This paper is structured as follows. Section II provides an overview of related work, which is followed by a discussion on depthwise separable convolutions. This is followed by a description of the DepthwiseGAN architecture in Section III and a description of the experiments conducted in Section IV. Finally, results and the conclusions are provided in Sections V and VI respectively.

## II. BACKGROUND

### A. Generative Adversarial Networks

GANs were first introduced by Goodfellow et al. in 2014 [6]. A GAN is composed of two neural networks namely, a generator  $G$  and a discriminator  $D$ , which are trained by playing a mini-max game. In the case of image translation, to learn a generator distribution  $p_g$  over data  $x$ , the generator creates a mapping function, parameterized by  $\theta_g$  from a prior latent distribution  $p_z(z)$  to data space  $G(z; \theta_g)$ . The discriminator  $D(x; \theta_d)$ , on the other hand, learns parameters  $\theta_d$  to distinguish whether images are from the training data or from the generator. The mini-max game function  $V(G, D)$  is expressed as follows:

$$\begin{aligned} \min_G \max_D V(D, G) &= \mathbb{E}_D + \mathbb{E}_G \quad (1) \\ \text{where } \mathbb{E}_D &= \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x)] \\ \mathbb{E}_G &= \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))] \end{aligned}$$

### B. Conditional GANs

Conditional GANs are an extension of the original GAN that includes additional information in both the generator and discriminator [17]. This allows for the ability to control the generated output image, so as to produce more flexible synthetic data. This information,  $y$ , is typically a label applied to the resulting output image, for example, skin complexion, emotional state or hair color (see Figure 1). The mini-max

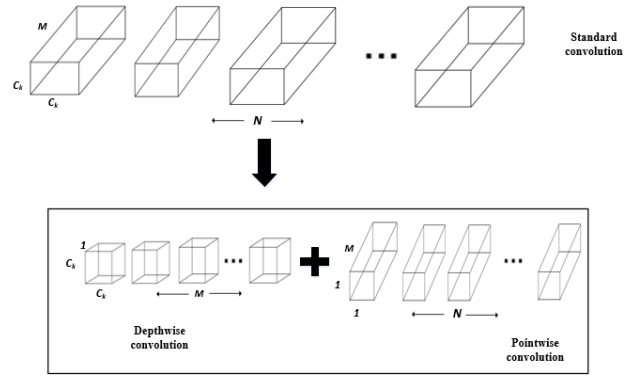


Figure 2. In contrast to traditional convolution operations performed in deep learning, depthwise separable convolution perform convolutions on subsets of the input tensor and aggregate the information across these subsets using a single pointwise convolution.

objective function from equation (2) can be expressed as follows for conditional GANs:

$$\begin{aligned} \min_G \max_D V(D, G) &= \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log D(x|y)] + \\ &\mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z|y)))] \quad (2) \end{aligned}$$

Conditional GANs can be applied in many fields where there is limited data and data augmentation is required.

### C. Depthwise Separable Convolutions

Depthwise separable convolutions were introduced by Chollet in 2016 as a replacement for standard Convolutions [15]. They are a form of factorized convolutions that apply separate convolution operations to every input channel of a tensor. Depthwise separable convolutions have fewer trainable parameters and have shown improved performance in image classification tasks [18]. Depthwise separable convolutions are an extension to the Xception layer [19] and are composed of two components, namely a depthwise convolution and a pointwise convolution. The depthwise convolution performs a spatial convolution over each input channel data while the pointwise convolution layer performs a  $1 \times 1$  convolution that merges information from the previous step across all channels. The original convolution layer dimensions can be expressed as  $(H, W, C_{in}, C_{out}, K)$  where  $H$  is the height of an input tensor,  $W$  is the width,  $C$  is the depth and  $K$  the size of the convolution kernel. A depthwise separable convolution operation can be expressed using dimensions  $(H, W, C_{in}, K)$  for the depthwise convolution and  $(H, W, C_{out})$  for the pointwise convolution. This results in an operation reduction of  $\frac{1}{C_{out}} + \frac{1}{K^2}$ . Figure 2 illustrates the comparison between the standard convolution and depthwise separable convolution architectures.

Depthwise separable convolution have found success in a variety of applications [20, 21, 22, 23]. The work of Nguyen and Ray proposed an adaptive convolution block method that learns the upsampling algorithm [24]. They replace traditional convolutions with depthwise separable convolutions in the

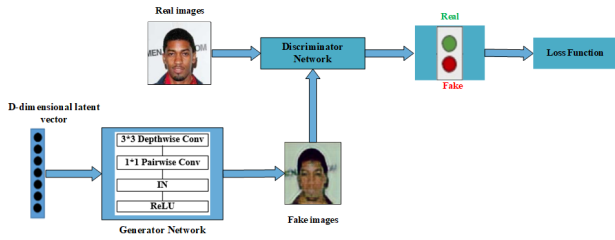


Figure 3. The DepthwiseGAN architecture is composed of two networks (Generator ( $G$ ) and Discriminator ( $D$ )). We used depthwise separable convolutions in the generator coupled with Instance Normalization (IN) and ReLU activation functions.

generator to improve the performance of a weak baseline model. Moreover, Wojna et al. have also applied depthwise separable convolutions in GAN architectures, where they introduced a bilinear additive upsampling layer, which improves performance [25]. However, our work investigates the performance of depthwise separable convolutions on the upsampling, downsampling and bottleneck layers. This is only applied in the generator. Our results show that depthwise separable convolutions are more effective when only used on the generator, that they reduce the number of parameters, which speeds up training time, but that deeper models are required to obtain similar performance to standard convolutions.

### III. DEPTHWISEGAN ARCHITECTURES

This section describes the proposed modifications made to StarGAN in order to produce lightweight networks with fewer trainable parameters on the generator network for experimental testing.

The architecture used for testing is adapted from [14], and consists of two networks, where the generator consists of a stride size of two for downsampling and 11 depthwise separable convolutions (see figure 3). Instance normalization is an operation of removing instance-specific contrast information from the content image which prevents mean and covariance shift and simplifies the learning process [26]. We applied instance normalization and ReLU activations after each pointwise convolution. For the discriminator network, a Markovian discriminator (PatchGAN) [27] was adopted because it is a fixed-size patch discriminator that is easily applied to  $256 \times 256$  images. We prepared three models for training, which we term DepthwiseDG, DepthwiseG, and DeeperDepthwiseG models. The DepthwiseDG model replaces convolution layers with depthwise convolutions in both the generator and discriminator, the DepthwiseG model replaces convolutions in only the generator, while the DeeperDepthwiseG model is a deeper model with 11 depthwise separable convolution layers in the generator. All three models were compared against the original StarGAN architecture.

### IV. EXPERIMENTAL METHODS

The models above were tested on a number of datasets. These are briefly described, in addition to a description of

the DepthwiseGAN training process. Finally, the Fréchet Inception Distance (FID), which aims to measure the quality of generated image distribution is described.

#### A. Datasets

Conditional generation of facial images was explored using three datasets. Each of these is discussed below.

**CelebA** The CelebFace Attributes dataset contains about 200k faces of celebrities [28]. Each image is annotated with 40 attributes and faces cover a relatively large pose variation, with background clutter included. Images were cropped, with the faces centered and resized to  $128 \times 128$  pixels. We used the seven facial domains investigated in [14] to train our models.

**Stirling 3D Face Database** The Stirling 3D Face dataset consist of 3,339 images collected from 99 participants (45 males and 54 females) [29]. Each participant makes seven facial expressions and images are captured at four different angles. The images were cropped to  $256 \times 256$  pixels, with the faces are centred and then resized to  $128 \times 128$ .

**RaFD** The Radboud Face Database (RaFD) consists of 49 people divided into two subsets, comprising 39 Caucasian Dutch adults (19 female and 20 male) and 10 Caucasian Dutch children (6 female and 4 male) [30]. The images include eight facial expressions captured from three different angles. Images were prepared in the same way as the Stirling 3D Face Database.

#### B. Training process

The DepthwiseGAN training process is illustrated in algorithm 1, and uses the following adversarial loss [14]:

$$\mathcal{L}_{adv} = \mathbb{E}_x[\log D_{src}(x)] + \mathbb{E}_{x,c}[\log(1 - D_{src}(G(x,c)))] \quad (3)$$

where  $G$  generates an image  $G(x, c)$  which is conditioned on both the input image  $x$  and the target domain label  $c$  and  $D_{src}(x)$  is a probability distribution over sources given by  $D$ . The modified objective function in [14] is also adapted and used to optimize both the generator  $G$  and discriminator  $D$  as follows:

$$\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls} \mathcal{L}_{cls}^r \quad (4)$$

$$\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{cls} \mathcal{L}_{cls}^f + \lambda_{rec} \mathcal{L}_{rec} \quad (5)$$

where  $\lambda_{cls}$  and  $\lambda_{rec}$  are hyper-parameters that trade-off domain classification and reconstruction losses.

All the parameters for the training followed the StarGAN training procedure and all training was conducted on a single NVIDIA Tesla K20c GPU. The Adam optimizer was used to train all the models.

---

**Algorithm 1 : Model Training**


---

**Input:** Given a set of real images  $\{x_1, x_2, \dots\} \sim p(x)$ ,  $\lambda_{cls} = 1, \lambda_{rec} = 10$

- 1: **for** number of training iterations **do**
  - 2:   Sample mini-batch of latent sample  $\{z\}$  from latent prior  $p_z(z)$
  - 3:   Randomly sample mini-batch of one sample  $\{x_1, x_2, y\}$  from training set.
  - 4:   Update the parameter  $\theta_d$  by descending its stochastic gradient with the Adam optimizer:
  - 5:    $\mathcal{L}_D = -\mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls}^r$
  - 6:   Update the generator  $\theta_g$  by ascending its stochastic gradient with Adam optimizer :
  - 7:    $\mathcal{L}_G = \mathcal{L}_{adv} + \lambda_{cls}\mathcal{L}_{cls}^f + \lambda_{rec}\mathcal{L}_{rec}$
  - 8: **end for**
  - 9: **Output** Trained generator  $G$  and discriminator  $D$ .
- 

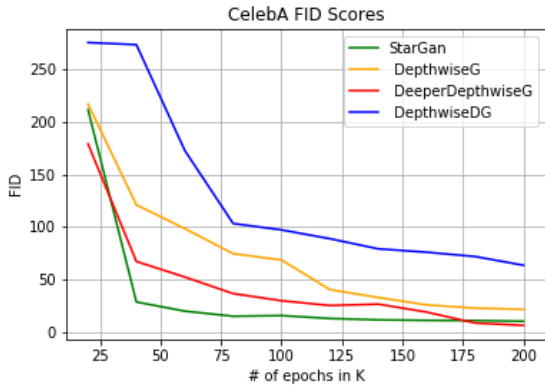


Figure 4. Scores for FID on four models that we experimented on and K represents thousands. DeeperDepthwiseG performs similarly to the baseline model. DepthwiseDG performed poor against all models.

### C. Fréchet Inception Distance

Images generated using GANs can be very realistic when viewed qualitatively, but metrics are required to evaluate performance quantitatively. Here, we rely on the Fréchet Inception Distance (FID) [31], which approximates the images distributions as multivariate Gaussians over pre-trained bottleneck features. The Fréchet distance is measured between two multivariate Gaussians as follows:

$$FID = \|\mu_r - \mu_g\|^2 + Tr(\Sigma_r + \Sigma_g - 2(\Sigma_r \Sigma_g)^{1/2}). \quad (6)$$

In this work,  $X_r \sim (\mu_r, \Sigma_r)$  and  $X_g \sim (\mu_g, \Sigma_g)$  are the mean and covariance of 2048-dimensional activations extracted from a pre-trained model (Inception-v3), computed using both the training data images and samples from the generator. We resized our images to  $128 \times 128$  dimensions in training phases. We sampled 140 images after every 20 000 epochs for FID evaluation.

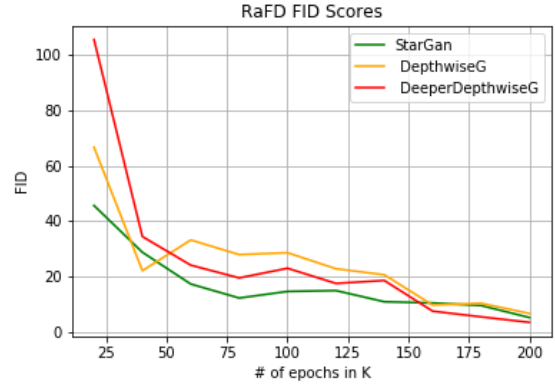


Figure 5. FID scores computed for RaFD dataset. DeeperDepthwiseG obtain similar performance to starGans, and all models exhibit similar convergence after 200 epochs.

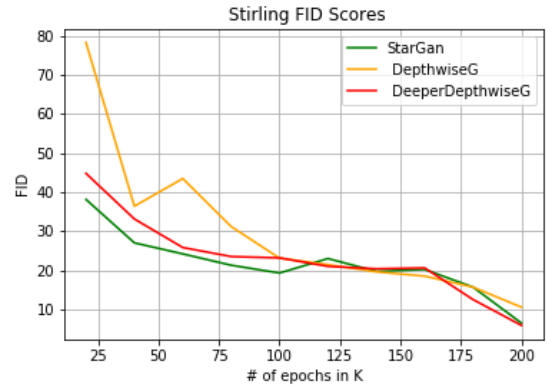


Figure 6. The FID scores for the Stirling dataset exhibit similar performance to that obtained on other sets.

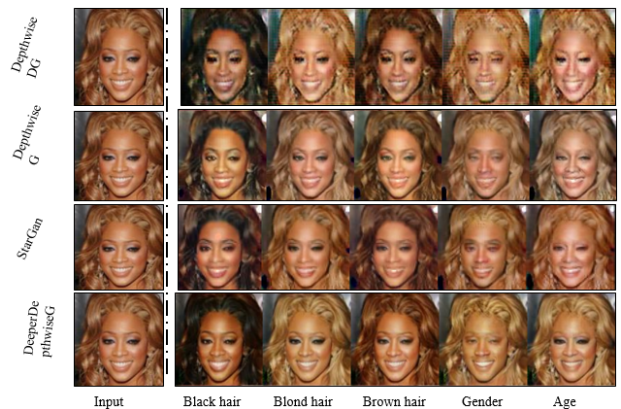


Figure 7. Applying StarGan and our three models for image-to-multi domain translation shows that having depthwise separable convolutions on the discriminator can have a bad effect on the results shown on top. While adding more depthwise separable convolutions on the generator results into realistic images are shown at the bottom of the figure.

## V. RESULTS

This section provides the experimental results obtained where we compared our models with the StarGan model in a multi-domain translation task. We used three datasets for our experiments and evaluated the quality of our results using the FID. Training the standard StarGan takes about three days (8.5M parameters for the generator and 44M parameters for discriminator). In contrast, training DepthwiseGANs is faster. The DepthwiseG model takes one day and 12 hours of training on a model with 1.5M parameters for generator and 44M parameters for discriminator, training takes two days for the DeeperDepthwiseG model with 5.6M parameters for generator and 44M parameters for discriminator, and 9 hours for the DepthwiseDG model with 1.5M parameters for generator and 32M parameters for discriminator.

### A. Results on CelebA

We compared DepthwiseGANs against StarGan model. Figure 7 shows the results of facial attribute transfer on the CelebA dataset using the four models. Results indicate that including depthwise convolutions in the discriminator leads to poor quality image generation. In contrast, restricting depthwise separable convolutions to the generator and increasing the depth yields results indistinguishable from StarGan. The figure shows that using depthwise separable convolutions in both networks degraded images, which appear almost cartoon-like. As a result, it appears that additional learning parameters are required to distinguish between image domains in the discriminator.

The FID measure in Figure 4 shows that the convergence of the distribution of images generated by models using depthwise separable convolutions to the training dataset distribution is faster for StarGans, although similar performance is obtained after about 175 epochs. It is clear that deeper models perform better, indicating that the model capacity introduced by additional parameters is important if generated images are to follow a similar distribution to that of the training set.

### B. Results on RaFD and Stirling

We also trained our models on RaRD and Stirling datasets for facial expression transfer. Both the Stirling and RaFD datasets comprise of similar facial expressions, captured in different environments. Results illustrated in Figure 8 shows that the DeeperDepthwiseG model generates high-quality images when compared to the original StarGan. The FID scores in Figures 5 and 6 show that the difference between generated and training image distributions is similar for all models. Qualitatively, images produced by the model with fewer parameters have blurry regions, and it appears that the model has limited learning capabilities due to the reduced parameter set. However, the model with more parameters generates better samples, highlighting the importance of model capacity further.

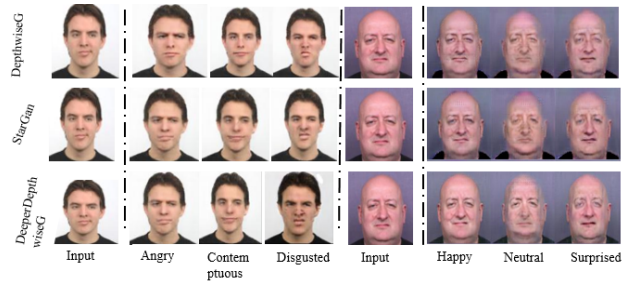


Figure 8. Samples generated show that generative models with additional capacity can perform well. The RaFD and Stirling datasets are similar, but captured under different lighting conditions.

## VI. CONCLUSION

In this paper, we investigated the effects of introducing depthwise separable convolutions to the StarGan architecture. Results showed that the generation of high-quality images relies on the depth or capacity of the network, and that GANs trained using depthwise separable convolutions require additional capacity to produce similar results to StarGans, although they do reduce computational requirements substantially.

## REFERENCES

- [1] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, “Unpaired image-to-image translation using cycle-consistent adversarial networks,” *arXiv preprint*, 2017.
- [2] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, “Photo-realistic single image super-resolution using a generative adversarial network.” in *CVPR*, vol. 2, no. 3, 2017, p. 4.
- [3] H. Zhang, T. Xu, H. Li, S. Zhang, X. Huang, X. Wang, and D. Metaxas, “Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks,” *arXiv preprint*, 2017.
- [4] R. Huang, S. Zhang, T. Li, R. He *et al.*, “Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis,” *arXiv preprint arXiv:1704.04086*, 2017.
- [5] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, “Image analogies,” in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 327–340.
- [6] 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*, 2014, pp. 2672–2680.
- [7] D. P. Kingma and P. Dhariwal, “Glow: Generative flow with invertible 1x1 convolutions,” *arXiv preprint arXiv:1807.03039*, 2018.
- [8] M.-Y. Liu, T. Breuel, and J. Kautz, “Unsupervised image-to-image translation networks,” in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.
- [9] T. Chavdarova and F. Fleuret, “Sgan: An alternative training of generative adversarial networks,” in *Proceedings*

- of the *IEEE international conference on Computer Vision and Pattern Recognition*, 2018.
- [10] N. Park, A. Anand, J. R. A. Moniz, K. Lee, T. Chakraborty, J. Choo, H. Park, and Y. Kim, "Mmgan: Manifold matching generative adversarial network for generating images," *arXiv preprint arXiv:1707.08273*, 2017.
- [11] B. Neyshabur, S. Bhojanapalli, and A. Chakrabarti, "Stabilizing gan training with multiple random projections," *arXiv preprint arXiv:1705.07831*, 2017.
- [12] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein gan," *arXiv preprint arXiv:1701.07875*, 2017.
- [13] R. D. Hjelm, A. P. Jacob, T. Che, A. Trischler, K. Cho, and Y. Bengio, "Boundary-seeking generative adversarial networks," *arXiv preprint arXiv:1702.08431*, 2017.
- [14] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," *arXiv preprint*, vol. 1711, 2017.
- [15] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *arXiv preprint*, pp. 1610–02 357, 2017.
- [16] E. J. Crowley, G. Gray, and A. Storkey, "Moonshine: Distilling with cheap convolutions," *arXiv preprint arXiv:1711.02613*, 2017.
- [17] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.
- [18] J. Shao, C. Qu, J. Li, and S. Peng, "A lightweight convolutional neural network based on visual attention for sar image target classification," *Sensors*, vol. 18, no. 9, p. 3039, 2018.
- [19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.
- [20] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [21] Ł. Kaiser, A. N. Gomez, and F. Chollet, "Depthwise Separable Convolutions for Neural Machine Translation," 2017.
- [22] G. Wang, G. Yuan, T. Li, and M. Lv, "An multi-scale learning network with depthwise separable convolutions," *Transactions on Computer Vision and Applications*, pp. 1–8, 2018.
- [23] M. Mahdianpari, B. Salehi, M. Rezaee, F. Mohammadianesh, and Y. Zhang, "Very Deep Convolutional Neural Networks for Complex Land Cover Mapping Using Multispectral Remote Sensing Imagery," *MDPI*, 2018.
- [24] N. M. Nguyen and N. Ray, "Generative adversarial networks using adaptive convolution," *arXiv preprint arXiv:1802.02226*, 2018.
- [25] Z. Wojna, V. Ferrari, S. Guadarrama, N. Silberman, L.-C. Chen, A. Fathi, and J. Uijlings, "The devil is in the decoder," *arXiv preprint arXiv:1707.05847*, 2017.
- [26] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization. arxiv 2016," *arXiv preprint arXiv:1607.08022*.
- [27] C. Li and M. Wand, "Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks," pp. 1–17, 2016.
- [28] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep Learning Face Attributes in the Wild," *ICCV*, pp. 3730–3738, 2015.
- [29] (2018) Psychological image collection at stirling (pics). Accessed 2018/09/10. [Online]. Available: <http://www.pics.stir.ac.uk>
- [30] O. Langner, R. Dotsch, G. Bijlstra, D. H. J. Wigboldus, S. T. Hawk, A. Van, O. Langner, R. Dotsch, G. Bijlstra, and D. H. J. Wigboldus, "Presentation and validation of the Radboud Faces Database," no. 907172236, 2010.
- [31] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in Neural Information Processing Systems*, 2017, pp. 6626–6637.