High-resolution Deep Convolutional Generative Adversarial Networks

J. D. Curtó*,1,2,3,4, I. C. Zarza*,1,2,3,4, F. Torre2,5, I. King[1], and M. R. Lyu[1].
[1]The Chinese University of Hong Kong. [2]Carnegie Mellon.
[3]Eidgenössische Technische Hochschule Zürich. [4]City University of Hong Kong. [5]Facebook.
*Both authors contributed equally.

Figure 1. **HDCGAN Synthetic Images**. A set of random samples. Our system generates high-resolution synthetic faces with an extremely high level of detail. HDCGAN goes from random noise to realistic synthetic pictures that can even fool humans. To demonstrate this effect, we create the Dataset of Curtó & Zarza, the first GAN augmented dataset of faces.

Generative Adversarial Networks (GANs) [Goodfellow et al. 2014] convergence in a high-resolution setting with a computational constrain of GPU memory capacity has been beset with difficulty due to the known lack of convergence rate stability. In order to boost network convergence of DCGAN (Deep Convolutional Generative Adversarial Networks) [Radford et al. 2016] and achieve good-looking high-resolution results we propose a new layered network, HDCGAN, that incorporates current state-of-the-art techniques for this effect. Glasses, a mechanism to arbitrarily improve the final GAN generated results by enlarging the input size by a telescope ζ is also presented. A novel bias-free dataset, Curtó & Zarza[1] [2], containing human faces from different ethnical groups in a wide variety of illumination conditions and image resolutions is introduced. Curtó is enhanced with HDCGAN synthetic images, thus being the first GAN augmented dataset of faces. We conduct extensive experiments on CelebA [Liu et al. 2015], CelebA-hq [Karras et al. 2018] and Curtó. HDCGAN is the current state-of-the-art in synthetic image generation on CelebA achieving a MS-SSIM of 0.1978 and a FRÉCHET Inception Distance of 8.44.

CCS Concepts: • **Neural Networks**;

Additional Key Words and Phrases: Generative Adversarial Network, Convolutional Neural Network, Synthetic Faces.

1 INTRODUCTION

Developing a Generative Adversarial Network (GAN) [Goodfellow et al. 2014] able to produce good quality high-resolution samples from images has important applications [Bousmalis et al. 2017; Chen et al. 2019; Li et al. 2017; Lombardi et al. 2018; Portenier et al. 2018; Romero et al. 2018; Sankaranarayanan et al. 2018; Wang et al. 2018a,b; Wu et al. 2016; Yang et al. 2017; Yu et al. 2018; Zhu et al. 2017] including image inpainting, 3D data, domain translation, video synthesis, image edition, semantic segmentation and semi-supervised learning.

In this paper, we focus on the task of face generation, as it gives GANs a huge space of learning attributes. In this context, we introduce the Dataset of Curtó & Zarza, a well-balanced collection of images containing 14,248 human faces from different ethnical groups and rich in a wide range of learnable attributes, such as gender and age diversity, hair-style and pose variation or presence of smile, glasses, hats and fashion items. We also ensure the presence of changes in illumination and image resolution. We propose to use Curtó as de facto approach to empirically test the distribution learned by a GAN, as it offers a challenging problem to solve, while keeping the number of samples, and therefore training time, bounded. It can also be used as a drop-in substitute of MNIST for simple tasks of classification, say for instance using labels of ethnicity, gender, age, hair style or smile. It ships with scripts in TensorFlow and Python that allow benchmarks of classification. A set of random samples can be seen in Figure 2.

Despite improvements in GANs training stability [Mescheder et al. 2017, 2018; Salimans et al. 2016] and specific-task design during the last years, it is still challenging to train GANs to generate high-resolution images due to the disjunction in the high dimensional pixel space between supports of the real image and implied model distributions [Arjovsky and Bottou 2017; Sønderby et al. 2017].

[1]Curtó is available at https://www.github.com/curto2/c/
[2]Code is available at https://www.github.com/curto2/graphics/

{curto,zarza,king,lyu}@cse.cuhk.edu.hk, ftorre@cs.cmu.edu
decurto.tw dezarza.tw.

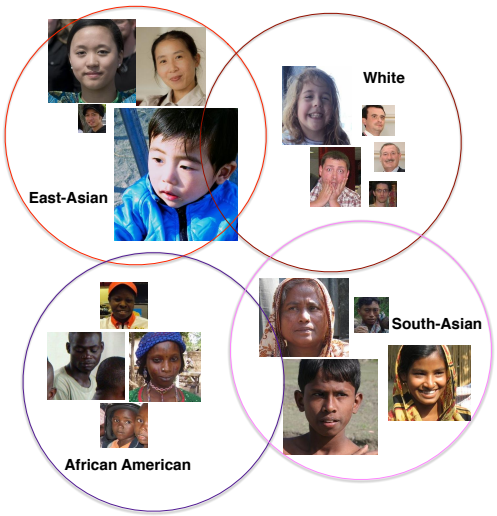

Figure 2. **Samples of Curtó**. A set of random instances for each class of ethnicity: African American, White, East-asian and South-asian. See Table 1 for numerics.

Our goal is to be able to generate indistinguishable sample instances using face data to push the boundaries of GAN image generation that scale well to high-resolution images (such as 512×512) and where context information is maintained.

In this sense, Deep Learning has a tremendous appetite for data. The question that arises instantly is, what if we were able to generate additional realistic data to aid learning using the same techniques that are later used to train the system. The first step would then be to have an image generation tool able to sample from a very precise distribution (e.g. faces from celebrities) which instances resemble or highly correlate with real sample images of the underlying true distribution. Once achieved, what is desirable and comes next is that these generated image points not only fit well into the original distribution set of images but also add additional useful information such as redundancy, different poses or even generate highly-probable scenarios that would be possible to see in the original dataset but are actually not present.

Current research trends link Deep Learning and Kernel Methods to establish a unifying theory of learning [Curtó et al. 2017]. The next frontier in GANs would be to achieve learning at scale with very few examples. To achieve the former goal this work contributes in the following:

- Network that achieves compelling results and scales well to the high-resolution setting where to the best of our knowledge the majority of other variants are unable to continue learning or fall into mode collapse.

- New dataset targeted for GAN training, Curtó, that introduces a wide space of learning attributes. It aims to provide a well-posed difficult task while keeping training time and resources tightly bounded to spearhead research in the area.

2 PRIOR WORK

Generative image generation is a key problem in Computer Vision and Computer Graphics. Remarkable advances have been made with the renaissance of Deep Learning. Variational Autoencoders (VAE) [Kingma and Welling 2014; Lombardi et al. 2018] formulate the problem with an approach that builds on probabilistic graphical models, where the lower bound of data likelihood is maximized. Autoregressive models (scilicet PIXELRNN [van den Oord et al. 2016]), based on modeling the conditional distribution of the pixel space, have also presented relative success generating synthetic images. Lately, Generative Adversarial Networks (GANs) [Antoniou et al. 2018; Goodfellow et al. 2014; Odena et al. 2017; Portenier et al. 2018; Radford et al. 2016; Wang and Gupta 2016; Zhu et al. 2016] have shown strong performance in image generation. However, training instability makes it very hard to scale to high-resolution (256×256 or 512×512) samples. Some current works on the topic pinpoint this specific problem [Zhang et al. 2017], where conditional image generation is also tackled while other recent techniques [Brock et al. 2019; Chen and Koltun 2017; Dosovitskiy and Brox 2016; Karras et al. 2018; Salimans et al. 2016; Wei et al. 2018; Zhao et al. 2017] try to stabilize training.

3 DATASET OF CURTÓ & ZARZA

Curtó contains 14,248 faces balanced in terms of ethnicity: African American, East-Asian, South-Asian and White. Mirror images are included to enhance pose variation and there is roughly 25% per image class. Attribute information, see Table 1, is composed of thorough labels of gender, age, ethnicity, hair color, hair style, eyes color, facial hair, glasses, visible forehead, hair covered and smile. There is also an extra set with 3,384 cropped labeled images of faces, ethnicity white, no mirror samples included, see Column 4 in Table 1 for statistics. We crawled Flickr to download images of faces from several countries that contain different hair-style variations and style attributes. These images were then processed to extract 49 facial landmark points using [Xiong and Torre 2013]. We ensure using Mechanical Turk that the detected faces are correct in terms of ethnicity and face detection. Cropped faces are then extracted to generate multiple resolution sources. Mirror augmentation is performed to further enhance pose variation.

Curtó introduces a difficult paradigm of learning, where different ethnical groups are present, with very varied fashion and hair styles. The fact that the photos are taken using non-professional cameras in a non-controlled environment, gives us multiple poses, illumination conditions and camera quality.

4 APPROACH

Generative Adversarial Networks (GANs) proposed by [Goodfellow et al. 2014] are based on two dueling networks, Figure 3; Generator G and Discriminator D. In essence, the process of learning consists of a two-player game where D tries to distinguish between the prediction of G and the ground truth, while at the same time G tries to fool D by producing fake instance samples as closer to the real ones as possible. The solution to a game is called NASH equilibrium.

Table 1. Dataset of Curtó & Zarza. Attribute Information. Descending order of class instances by number of samples, Column 3.

Attribute	Class	# Samples	# Extra
Age	Early Adulthood	3606	966
	Middle Aged	2954	875
	Teenager	2202	178
	Adult	1806	565
	Kid	1706	85
	Senior	1102	402
	Retirement	436	218
	Baby	232	14
Ethnicity	African American	4348	0
	White	3442	3384
	East Asian	3244	0
	South Asian	3214	0
Eyes Color	Brown	9116	2119
	Other	4136	875
	Blue	580	262
	Green	416	128
Facial Hair	No	12592	2821
	Light Mustache	466	156
	Light Goatee	444	96
	Light Beard	258	142
	Thick Goatee	168	39
	Thick Beard	166	68
	Thick Mustache	154	62
Gender	Male	7554	1998
	Female	6694	1386
Glasses	No	12576	2756
	Eyeglasses	1464	539
	Sunglasses	208	89
Hair Color	Black	8402	964
	Brown	3038	1241
	Other	1554	253
	Blonde	616	543
	White	590	347
	Red	48	36
Hair Covered	No	12292	3060
	Turban	1206	76
	Cap	722	237
	Helmet	28	11
Hair Style	Short Straight	5038	1642
	Long Straight	2858	857
	Short Curly	2524	287
	Other	2016	249
	Bald	1298	187
	Long Curly	514	162
Smile	Yes	8428	2118
	No	5820	1266
Visible Forehead	Yes	11890	3033
	No	2358	351

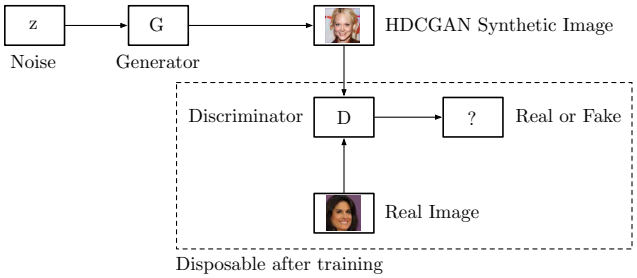

Figure 3. **Generative Adversarial Networks**. A two-player game between the Generator G and the Discriminator D. The dotted line denotes elements that will not be further used after the game stops, namely, end of training.

The min-max game entails the following objective function

$$\min_G \max_D V(D,G) = \mathbb{E}_{x\sim p_{data}}\left[\log D(x)\right] + \tag{1}$$

$$+\mathbb{E}_{z\sim p_z}\left[\log(1-D(G(z)))\right], \tag{2}$$

where x is a ground truth image sampled from the true distribution p_{data}, and z is a noise vector sampled from p_z (that is, uniform or normal distribution). G and D are parametric functions where $G : p_z \rightarrow p_{data}$ maps samples from noise distribution p_z to data distribution p_{data}.

The goal of the Discriminator is to minimize

$$L^{(D)} = -\frac{1}{2}\mathbb{E}_{x\sim p_{data}}\left[\log D(x)\right] - \tag{3}$$

$$-\frac{1}{2}\mathbb{E}_{z\sim p_z}\left[\log(1-D(G(z)))\right]. \tag{4}$$

If we differentiate it w.r.t $D(x)$ and set the derivative equal to zero, we can obtain the optimal strategy

$$D(x) = \frac{p_{data}(x)}{p_z(x)+p_{data}(x)}. \tag{5}$$

Which can be understood intuitively as follows. Accept an input, evaluate its probability under the distribution of the data, p_{data}, and then evaluate its probability under the generator's distribution of the data, p_z. Under the condition in D of enough capacity, it can achieve its optimum. Note the discriminator does not have access to the distribution of the data but it is learned through training. The same applies for the generator's distribution of the data. Under the condition in G of enough capacity, then it will set $p_z = p_{data}$. This results in $D(x) = \frac{1}{2}$, that is actually the NASH equilibrium. In this situation, the generator is a perfect generative model, sampling from $p(x)$.

As an extension to this framework, DCGAN [Radford et al. 2016] proposes an architectural topology based on Convolutional Neural Networks (CNNs) to stabilize training and re-use state-of-the-art

networks from tasks of classification. This direction has recently received lots of attention due to its compelling results in supervised and unsupervised learning. We build on this to propose a novel DCGAN architecture to address the problem of high-resolution image generation. We name this approach HDCGAN.

4.1 HDCGAN

Despite the undoubtable success, GANs are still arduous to train, particularly when we use big images (e.g. 512×512). It is very common to see D beating G in the process of learning, or the reverse, ending in unrecognizable imagery, also known as mode collapse. Only when stable learning is achieved, the GAN structure is able to succeed in getting better and better results with time.

This issue is what drives us to carefully derive a simple yet powerful structure that leverages common problems and gets a stable and steady training mechanism.

Self-normalizing Neural Networks (SNNs) were introduced in [Klambauer et al. 2017]. We consider a neural network with activation function f, connected to the next layer by a weight matrix $\mathbf{W}$, and whose inputs are the activations from the preceding layer x, $y = f(\mathbf{W}x)$.

We can define a mapping g that maps mean and variance from one layer to mean and variance of the following layer

$$\begin{pmatrix} \mu \\ \nu \end{pmatrix} \longmapsto \begin{pmatrix} \tilde{\mu} \\ \tilde{\nu} \end{pmatrix} : \begin{pmatrix} \tilde{\mu} \\ \tilde{\nu} \end{pmatrix} = g\begin{pmatrix} \mu \\ \nu \end{pmatrix}. \tag{6}$$

Common normalization tactics such as batch normalization ensure a mapping g that keeps (μ, ν) and $(\tilde{\mu}, \tilde{\nu})$ close to a desired value, normally $(0, 1)$.

SNNs go beyond this assumption and require the existence of a mapping $g : \Omega \longmapsto \Omega$ that for each activation y maps mean and variance from one layer to the next layer and at the same time have a stable and attracting fixed point depending on (ω, τ) in Ω. Moreover, the mean and variance remain in the domain Ω and when iteratively applying the mapping g, each point within Ω converges to this fixed point. Therefore, SNNs keep activations normalized when propagating them through the layers of the network.

Here (ω, τ) are defined as follows. For n units with activation x_c, $1 \leq c \leq n$ in the lower layer, we set n times the mean of the weight vector $w \in \mathbb{R}^n$ as $\omega := \sum_{c=1}^{n} w_c$ and n times the second moment as $\tau := \sum_{c=1}^{n} w_c^2$.

Scaled Exponential Linear Units (SELU) [Klambauer et al. 2017] is introduced as the choice of activation function in Feed-forward Neural Networks (FNNs) to construct a mapping g with properties that lead to SNNs.

$$selu(x) = \lambda \begin{cases} x & \text{if } x > 0 \\ \alpha \exp^x - \alpha & \text{if } x \leq 0. \end{cases} \tag{7}$$

Empirical observation leads us to say that the use of SELU greatly improves the convergence speed on the DCGAN structure, however, after some iterations mode collapse and gradient explosion completely destroy training when using high-resolution images. We conclude that although SELU gives theoretical guarantees as the optimal activation function in FNNs, numerical errors in the GPU computation degrade its performance in the overall min-max game of DCGAN. To alleviate this problem, we propose to use SELU and BatchNorm [Ioffe and Szegedy 2015] together. The motivation is that when numerical errors move $(\tilde{\mu}, \tilde{\nu})$ away from the attracting point that depends on $(\omega, \tau) \in \Omega$, BatchNorm will ensure it is close to a desired value and therefore maintain the convergence rate.

Experiments show that this technique stabilizes training and allows us to use fewer GPU resources, having steady diminishing errors in G and D. It also accelerates convergence speed by a great factor, as can be seen after some few epochs of training on CelebA in Figure 8.

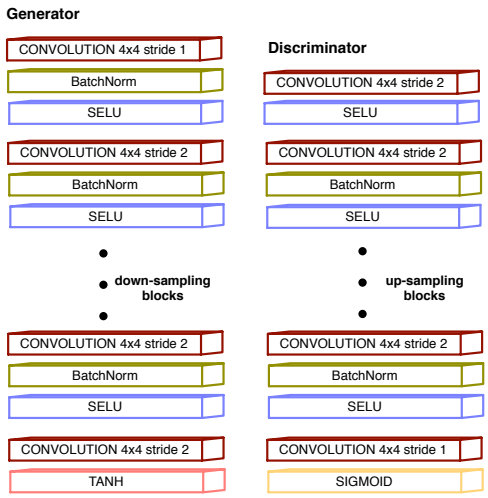

Figure 4. **HDCGAN Architecture**. Generator and Discriminator.

As SELU + BatchNorm (BS) layers keep mean and variance close to $(0, 1)$ we get an unbiased estimator of p_{data} with contractive finite variance. These are very desirable properties from the point of view of an estimator as we are iteratively looking for a MVU (Minimum Variance Unbiased) criterion and thus solving MSE (Minimum Square Error) among unbiased estimators. Hence, if the MVU estimator exists and the network has enough capacity to actually find the solution, given a sufficiently large sample size by the Central Limit Theorem, we can attain NASH equilibrium.

HDCGAN Architecture is described in Figure 4. It differs from traditional DCGAN in the use of BS layers instead of ReLUs.

We observe that when having difficulty in training DCGAN, it is always better to use a fixed learning rate and instead increase the batch size. This is because having more diversity in training, gives a steady diminishing loss and better generalization. To aid learning,

noise following a Normal $N(0,1)$ is added to both the inputs of D and G. We see that this helps overcome mode saturation and collapse whereas it does not change the distribution of the original data.

We empirically show that the use of BS induces SNNs properties in the GAN structure, and thus makes learning highly robust, even in the stark presence of noise and perturbations. This behavior can be observed when the zero-sum game problem stabilizes and errors in D and G jointly diminish, Figure 9. Comparison to traditional DCGAN, Wasserstein GAN [Arjovsky et al. 2017] and WGAN-GP [Gulrajani et al. 2017] is not possible, as to date, the majority of former methods, such as [Denton et al. 2015], cannot generate recognizable results in image size 512×512, 24GB GPU memory setting.

Thus, HDCGAN pushes up state-of-the-art results beating all former DCGAN-based architectures and shows that, under the right circumstances, BS can solve the min-max game efficiently.

4.2 Glasses

We introduce here a key technique behind the success of HDCGAN. Once we have a good convergence mechanism for large input samples, that is a concatenation of BS layers, we observe that we can arbitrarily improve the final results of the GAN structure by the use of a Magnifying Glass approach. Assuming our input length is $N \times M$, we can enlarge it by a constant factor, $\zeta_1 N \times \zeta_2 M$, which we call telescope, and then feed it into the network, maintaining the size of the convolutional filters untouched. This simple procedure works similar to how contact lenses correct or assist defective eyesight on humans and empowers the GAN structure to appreciate the inner properties of samples.

Note that as the input gets bigger so does the neural network. That is, the number of layers is implicitly set by the image size, see up-sampling and down-sampling blocks in Figure 4. For example, for an input size of 32 we have 4 layers while for an input size of 256 we have 7 layers.

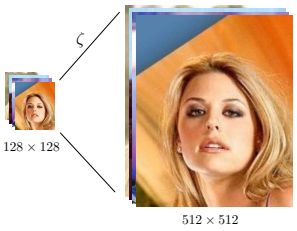

Figure 5. **Glasses on a set of samples from CelebA**. HDCGAN introduces the use of a Magnifying Glass approach, enlarging the input size by a telescope ζ.

We can empirically observe that BS layers together with Glasses induce high capacity into the GAN structure so that a NASH equilibrium can be reached. That is to say, the generator draws samples from p_{data}, which is the distribution of the data, and the discriminator is not able to distinguish between them, $D(x) = \frac{1}{2} \forall x$.

5 EMPIRICAL ANALYSIS

We build on DCGAN and extend the framework to train with high-resolution images using Pytorch. Our experiments are conducted using a fixed learning rate of 0.0002 and ADAM solver [Kingma and Ba 2015] with batch size 32 and 512×512 samples with the number of filters of G and D equal to 64.

In order to test generalization capability, we train HDCGAN in the newly introduced Curtó, CelebA and CelebA-hq.

Technical Specifications: 2 × NVIDIA Titan X, Intel Core i7-5820k@3.30GHz.

5.1 Curtó

The results after 150 epochs are shown in Figure 6. We can see that HDCGAN captures the underlying features that represent faces and not only memorizes training examples. We retrieve nearest neighbors to the generated images in Figure 7 to illustrate this effect.

Figure 6. **HDCGAN Example Results. Dataset of Curtó & Zarza**. 150 epochs of training. Image size 512×512.

Figure 7. **Nearest Neighbors. Dataset of Curtó & Zarza**. Generated samples in the first row and their five nearest neighbors in training (rows 2-6).

5.2 CelebA

CelebA is a large-scale dataset with 202,599 celebrity faces. It mainly contains frontal portraits and is particularly biased towards groups of ethnicity white. The fact that it presents very controlled illumination settings and good photo resolution, makes it a considerably easier problem than Curtó. The results after 19 epochs of training are shown in Figure 8.

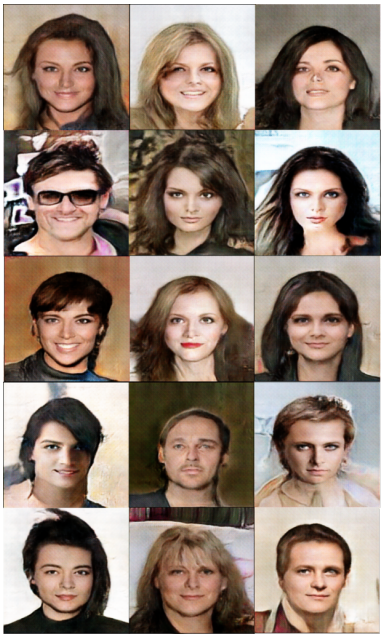

Figure 8. **HDCGAN Example Results. CelebA**. 19 epochs of training. Image size 512×512. The network learns swiftly a clear pattern of the face.

In Figure 9 we can observe that BS stabilizes the zero-sum game, where errors in D and G concomitantly diminish. To show the validity of our method, we enclose Figure 10, presenting a large number of samples for epoch 39. We also attach a zoomed-in example to appreciate the quality and size of the generated samples, Figure 11. Failure cases can be observed in Figure 12.

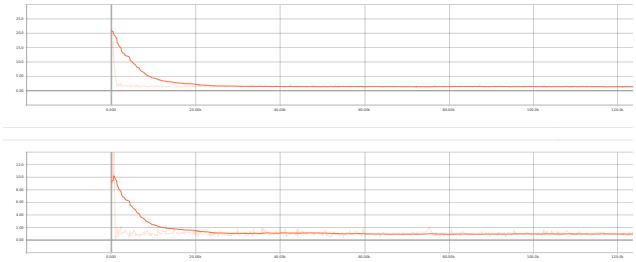

Figure 9. **HDCGAN on CelebA**. Error in Discriminator (top) and Error in Generator (bottom). 19 epochs of training.

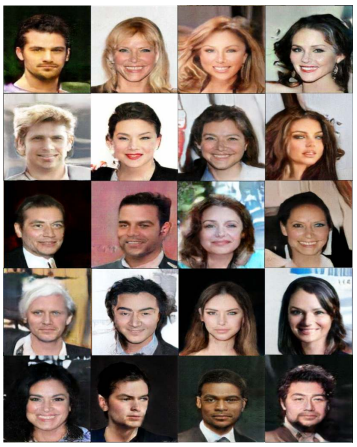

Figure 10. **HDCGAN Example Results. CelebA**. 39 epochs of training. Image size 512×512. The network generates distinctly accurate and assorted faces, including exhaustive details.

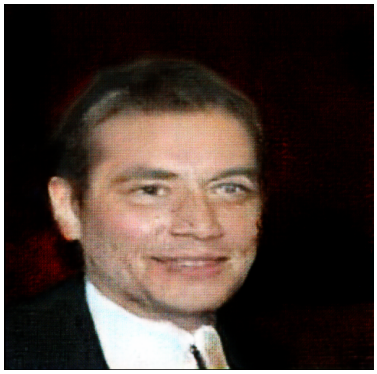

Figure 11. **HDCGAN Example Result. CelebA**. 39 epochs of training. Image size 512×512. 27% of full-scale image.

Figure 12. **HDCGAN Example Results. CelebA**. 39 epochs of training. Image size 512×512. Failure cases. The number of failure cases declines over time, and when present, they are of more meticulous nature.

Besides, to illustrate how fundamental our approach is, we enlarge Curtó with 4,239 unlabeled synthetic images generated by HDCGAN on CelebA, a random set can be seen in Figure 13.

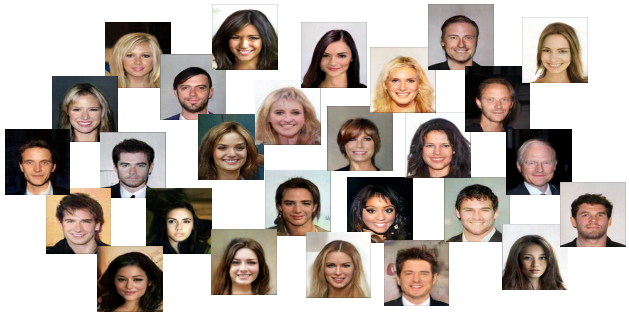

Figure 13. **HDCGAN Synthetic Images**. A set of random samples.

5.3 CelebA-hq

[Karras et al. 2018] introduces CelebA-hq, a set of 30.000 high-definition images to improve training on CelebA. A set of samples generated by HDCGAN on CelebA-hq can be seen in Figures 1, 14 and 15.

Figure 14. **HDCGAN Example Results. CelebA-hq**. 229 epochs of training. Image size 512×512. The network generates superior faces, with great attention to detail and quality.

Figure 15. **HDCGAN Example Result. CelebA-hq**. 229 epochs of training. Image size 512×512. 27% of full-scale image.

To exemplify that the model is generating new bona fide instances instead of memorizing samples from the training set, we retrieve nearest neighbors to the generated images in Figure 16.

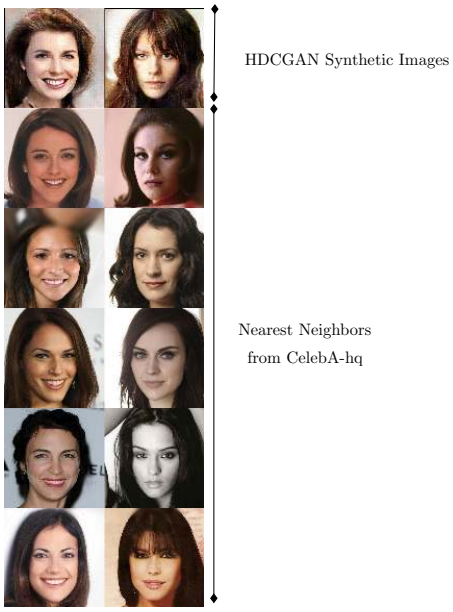

Figure 16. **Nearest Neighbors. CelebA-hq**. Generated samples in the first row and their five nearest neighbors in training (rows 2-6).

6 ASSESSING THE DISCRIMINABILITY AND QUALITY OF GENERATED SAMPLES

We build on previous image similarity metrics to qualitatively evaluate generated samples of generative models. The most effective of these is multi-scale structural similarity (MS-SSIM) [Odena et al. 2017]. We make comparison at resized image size 128×128 on CelebA. MS-SSIM results are averaged from 10,000 pairs of generated samples. Table 2 shows HDCGAN significantly improves state-of-the-art results.

Table 2. Multi-scale structural similarity (MS-SSIM) results on CelebA at resized image size 128×128. Lower is better.

	MS-SSIM
[Gulrajani et al. 2017]	0.2854
[Karras et al. 2018]	0.2838
HDCGAN	**0.1978**

We monitor MS-SSIM scores across several epochs averaging from 10,000 pairs of generated images to see the temporal performance, Figure 17. HDCGAN improves the quality of the samples while increases the diversity of the generated distribution.

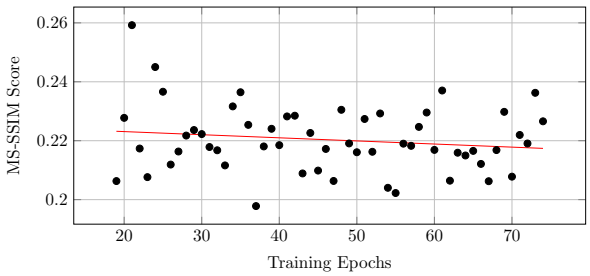

Figure 17. **MS-SSIM Scores on CelebA across several epochs**. Results are averaged from 10,000 pairs of generated images from epoch 19 to 74. Comparison is made at resized image size 128×128. Affine interpolation is shown in red.

In [Heusel et al. 2017] they propose to evaluate GANs using the FRÉCHET Inception Distance, which assesses the similarity between two distributions by the difference of two Gaussians. We make comparison at resized image size 64×64 on CelebA. Results are computed from 10,000 512×512 generated samples from epochs 36 to 52, resized at image size 64×64 yielding a value of 8.44, Table 3, clearly outperforming current reported scores in DCGAN architectures [Wu et al. 2018].

Table 3. FRÉCHET Inception Distance on CelebA at resized image size 64×64. Lower is better.

	Fréchet
[Karras et al. 2018]	16.3
[Wu et al. 2018]	16.0
HDCGAN	**8.44**

7 DISCUSSION

In this paper, we propose High-resolution Deep Convolutional Generative Adversarial Networks (HDCGAN) by stacking SELU + BatchNorm (BS) layers. The proposed method generates high-resolution images (e.g. 512×512) in circumstances where the majority of former methods fail. It exhibits a steady and smooth mechanism of training. It also introduces Glasses, the notion that enlarging the input image by a telescope ζ while keeping all convolutional filters unchanged, can arbitrarily improve the final generated results. HDCGAN is the current state-of-the-art in synthetic image generation on CelebA (MS-SSIM 0.1978 and FRÉCHET Inception Distance 8.44).

Further, we present a bias-free dataset of faces containing well-balanced ethnical groups, Curtó & Zarza, that poses a very difficult challenge and is rich on learning attributes to sample from. Moreover, we enhance Curtó with 4,239 unlabeled synthetic images generated by HDCGAN, being therefore the first GAN augmented dataset of faces.

REFERENCES

A. Antoniou, A. Storkey, and H. Edwards. 2018. Data Augmentation Generative Adversarial Networks. *ICLR* (2018).
M. Arjovsky and L. Bottou. 2017. Towards Principled Methods for Training Generative Adversarial Networks. *ICLR* (2017).
M. Arjovsky, S. Chintala, and L. Bottou. 2017. WASSERSTEIN Generative Adversarial Networks. *ICML* (2017).
K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan. 2017. Unsupervised Pixel-level Domain Adaptation with Generative Adversarial Networks. *IEEE International Conference on Computer Vision* (2017).
A. Brock, J. Donahue, and K. Simonyan. 2019. Large Scale GAN Training for High Fidelity Natural Image Synthesis. *ICLR* (2019).
Q. Chen and V. Koltun. 2017. Photographic Image Synthesis with Cascaded Refinement Networks. *IEEE International Conference on Computer Vision* (2017).
Y. Chen, W. Li, X. Chen, and L. Gool. 2019. Learning Semantic Segmentation from Synthetic Data: a Geometrically Guided Input-output Adaptation Approach. *IEEE International Conference on Computer Vision* (2019).
J. D. Curtó, I. C. Zarza, F. Yang, Alexander Smola, F. Torre, C. Ngo, and L. Gool. 2017. McKernel: a Library for Approximate Kernel Expansions in Log-linear Time. *arXiv:1702.08159* (2017).
E. Denton, S. Chintala, A. Szlam, and R. Fergus. 2015. Deep Generative Image Models Using a LAPLACIAN Pyramid of Adversarial Networks. *NIPS* 28 (2015).
A. Dosovitskiy and T. Brox. 2016. Generating Images with Perceptual Similarity Metrics Based on Deep Networks. *NIPS* (2016).
I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. 2014. Generative Adversarial Networks. *NIPS* 27 (2014).
I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville. 2017. Improved Training of WASSERSTEIN GANs. *NIPS* (2017).
M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. 2017. GANs Trained by a Two Time-scale Update Rule Converge to a Local NASH Equilibrium. *NIPS* 30 (2017).
S. Ioffe and C. Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *ICML* 37 (2015).
T. Karras, T. Aila, S. Laine, and J. Lehtinen. 2018. Progressive Growing of GANs for Improved Quality, Stability, and Variation. *ICLR* (2018).
D. Kingma and J. Ba. 2015. A Method for Stochastic Optimization. *ICLR* (2015).
D. P. Kingma and M. Welling. 2014. Auto-encoding Variational Bayes. *ICLR* (2014).
G. Klambauer, T. Unterthiner, A. Mayr, and S. Hochreiter. 2017. Self-normalizing Neural Networks. *NIPS* (2017).
C. Li, K. Xu, J. Zhu, and B. Zhang. 2017. Triple Generative Adversarial Nets. *NIPS* 30 (2017).
Z. Liu, P. Luo, X. Wang, and X. Tang. 2015. Deep Learning Face Attributes in the Wild. *ICCV* (2015).
S. Lombardi, J. Saragih, T. Simon, and Y. Sheikh. 2018. Deep Appearance Models for Face Rendering. *ACM Transactions on Graphics (SIGGRAPH)* 37 (2018).
L. Mescheder, S. Nowozin, and A. Geiger. 2017. The Numerics of GANs. *NIPS* (2017).
L. Mescheder, S. Nowozin, and A. Geiger. 2018. Which Training Methods for GANs Do Actually Converge? *ICML* (2018).
A. Odena, C. Olah, and J. Shlens. 2017. Conditional Image Synthesis with Auxiliary Classifier GANs. *ICML* (2017).
T. Portenier, Q. Hu, A. Szabó, S. A. Bigdeli, P. Favaro, and M. Zwicker. 2018. FaceShop: Deep Sketch-based Face Image Editing. *ACM Transactions on Graphics (SIGGRAPH)* 37 (2018).
A. Radford, L. Metz, and S. Chintala. 2016. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *ICLR* (2016).
A. Romero, P. Arbeláez, L. Gool, and R. Timofte. 2018. SMIT: Stochastic Multi-label Image-to-image Translation. *arXiv:1812.03704* (2018).
T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. 2016. Improved Techniques for Training GANs. *NIPS* 29 (2016).
S. Sankaranarayanan, Y. Balaji, A. Jain, S. N. Lim, and R. Chellappa. 2018. Learning from Synthetic Data: Addressing Domain Shift for Semantic Segmentation. *IEEE International Conference on Computer Vision* (2018).
C. Sønderby, J. Caballero, L. Theis, W. Shi, and F. Huszár. 2017. Amortised MAP Inference for Image Super-resolution. *ICLR* (2017).
A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu. 2016. Pixel Recurrent Neural Networks. *ICML* (2016).
T. Wang, M. Liu, J. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro. 2018a. Video-to-video Synthesis. *NIPS* (2018).
T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro. 2018b. High-resolution Image Synthesis and Semantic Manipulation with Conditional GANs. *IEEE International Conference on Computer Vision* (2018).
X. Wang and A. Gupta. 2016. Generative Image Modeling Using Style and Structure Adversarial Networks. *EUROPEAN Conference on Computer Vision* (2016).
X. Wei, B. Gong, Z. Liu, W. Lu, and L. Wang. 2018. Improving the Improved Training of WASSERSTEIN GANs: a Consistency Term and its Dual Effect. *ICLR* (2018).
J. Wu, Z. Huang, J. Thoma, D. Acharya, and L. Gool. 2018. WASSERSTEIN Divergence for GANs. *EUROPEAN Conference on Computer Vision* (2018).
J. Wu, C. Zhang, T. Xue, W. T. Freeman, and J. B. Tenenbaum. 2016. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-adversarial Modeling.

NIPS 29 (2016).

X. Xiong and F. Torre. 2013. Supervised Descent Method and its Application to Face Alignment. *CVPR* (2013).

C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li. 2017. High-resolution Image Inpainting Using Multi-scale Neural Patch Synthesis. *IEEE International Conference on Computer Vision* (2017).

J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. 2018. Generative Image Inpainting with Contextual Attention. *IEEE International Conference on Computer Vision* (2018).

H. Zhang, T. Xu, H. Li, S. Zhang, X. Wang, X. Huang, and D. Metaxas. 2017. Stack-GAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. *IEEE International Conference on Computer Vision* (2017).

J. Zhao, M. Mathieu, and Y. LeCun. 2017. Energy-based Generative Adversarial Networks. *ICLR* (2017).

J. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. 2016. Generative Visual Manipulation on the Natural Image Manifold. *EUROPEAN Conference on Computer Vision* (2016).

J. Zhu, T. Park, P. Isola, and A. A. Efros. 2017. Unpaired Image-to-Image Translation Using Cycle-consistent Adversarial Networks. *IEEE International Conference on Computer Vision* (2017).