HUMAN BLASTOCYST IMAGE GENERATION USING GENERATIVE ADVERSARIAL NETWORKS

Evangelos Tikas[⋆] *Lazaros Alexios Iliadis*[⋆] *Sotirios Sotiroudis*[⋆] *Achilles Boursianis*[⋆]
Konstantinos-Iraklis D. Kokkinidis[†] *Achilleas Papatheodorou*[‡] *Sotirios K. Goudos* [⋆,§]

[⋆] ELEDIA@AUTH, School of Physics, Aristotle University of Thessaloniki, Thessaloniki, Greece
[§] Department of Electronics and Communication Engineering, Bharath University, Chennai, India
[†] Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece
[‡] Embryolab Fertility Clinic, Thessaloniki, Greece

ABSTRACT

Several medical fields such as In Vitro Fertilization (IVF) suffer from limited databases d require augmentations, to be suitable for machine learning (ML) tasks such as classification. We test the use of various Generative Adversarial Network (GAN) architectures to tackle the problem of limited data by generating high-quality samples. We compare two GAN-based data augmentation techniques for a limited medical image database, utilizing synthetic images. A GAN can create synthetic images by learning to model a probability density function that can represent large data like images. Synthetic samples are then evaluated using metrics and visual evaluation with respect to the original "real" samples.

Index Terms— In-Vitro Fertilization, Generative Adversarial Networks, Generative AI, Synthetic Images, Deep Learning, Machine Learning

1. INTRODUCTION

In recent years significant progress has been made in the field of synthetic image generation [1], [2], [3]. Advanced neural network architectures, and in-depth research on generative models, have been vital for the evolution of models such as GANs [1], Variational Auto-Encoders (VAE) [4] etc. Artificial data and the concept of Generative AI have proven fruitful in bringing solutions in various fields such as automotive, physics, and medical applications. IVF is one of the fields which can benefit from artificial data (e.g. artificial blastocyst images) to surpass the problem of reduced databases.

IVF is one of the most effective assisted reproductive approaches and a treatment for infertility. However, it involves a series of costly and complicated procedures. Furthermore, the succession rate of this approach is $\sim 30\%$. To increase the succession rate, while keeping the procedure simple, it is necessary to rely on new technological concepts, such as deep learning (DL). Deep neural networks (DNN) have become one of the most popular modern tools for image analysis and classification [5] . DL-empowered IVF applications can be achieved through the use of Image Processing and Artificial Neural Networks (ANN) in human blastocyst images. To successfully employ DL algorithms in IVF databases for analysis, classification, or any other task, the problem of data scarcity must be addressed.

Traditional augmentation techniques, such as image transformations, are a valid choice but often lack variety and realism. GAN-generated images offer realistic, real-life features in images that can deceive the human observer, or even other DL models which can in turn learn to generalize their results based on the combined use of artificial and synthetic data. In human blastocysts, this will require generating data that represent real-looking embryo images. Successful generation means that the model has all the necessary attributes of a real image to provide a fully realistic synthetic version [6].

We examine two different GAN architectures, namely, the Progressive GAN (ProGAN or PGAN) [7] and the StyleGAN [2]. Both approaches have achieved good results.

In this paper, one instance for each architecture was chosen to produce synthetic images for human blastocysts. We compare the loss functions of those networks as well as the quality of image samples. Finally, we provide mathematical metrics to quantify the resulting fidelity, suitable for GAN-based architectures.

2. METHODS

Our hypothesis is based on the fact that in medical samples obtaining large amounts of genuine data is often difficult or unethical [8]. This is a major challenge in the field of medical and healthcare research. GANs are generative AI models that employ an adversarial game, with two networks pitched against each other, to generate samples that resemble the original ones.

GANs are comprised of two separate models, a Generator, which is a generative model, and a discriminator, a discriminative model. The generative model is compared to an adversary: a discriminative model that learns to determine whether

a sample is from the model distribution p_z or the data distribution p_{data} [1]. The generative model must learn to produce samples (e.g. images) from a latent code, close to the data distribution, deceiving the Discriminator to the point that "fake" data are indistinguishable from the "real" ones. Training continues until this goal is achieved.

Usually models, share some common design blocks, most prominently convolution layers, from CNNs [9]. Convolutions are effective at filtering data with spatial characteristics. Training with backpropagation is used on both models and forward propagation is used only on the generator to generate artificial samples. The original paper [1] proposes the two-player minimax game, loss function for GANs:

$$\min_G \max_D V(D,G) = E_{x=p_{data}}(x)[logD(x)] + E_{z=p_z(z)}[log(1-D(G(z)))] \quad (1)$$

where x is sampled from data distribution p_{data} and z is the latent vector. The first term determines the ratio to maximize the Discriminator output, while the second term determines the output of D, fed by a generated sample $G(z)$, where the objective is to minimize $log(1-D(G(z)))$.

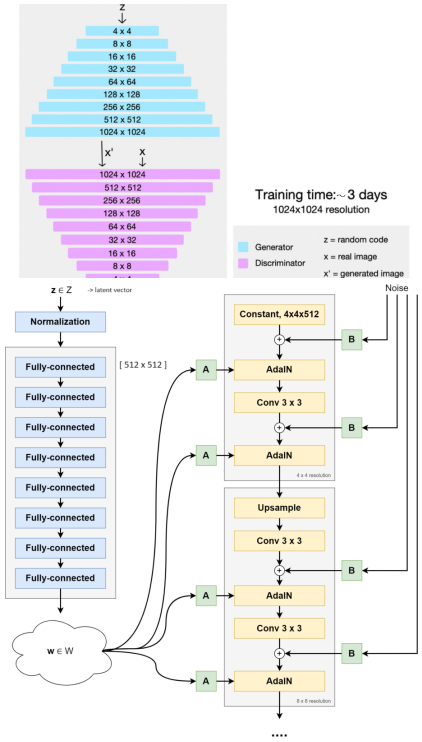


Fig. 1. Example of GAN models, ProGAN (top), StyleGAN (bottom) implemented in this study.

First, we analyze the architecture proposed by NVIDIA as ProGAN [7]. ProGAN reinstated the use of GANs for high-resolution images. The term "progressive" means that training starts with low-resolution images and then progressively increases the resolution by adding layers to the networks. In a way, the network capacity is increased overtime, for both the Discriminator and the Generator, regularizing the training process for improved stability before reaching higher resolutions. The generator and discriminator networks are mirror images of each other and always grow in synchronization. All layers remain trainable throughout the training process. When new layers are added to the networks, a smooth fade-in is introduced to avoid transition shocks from the already trained lower layers to the new ones.

ProGAN employs normalization using a mini-batch standard deviation for each feature in each spatial location over a mini-batch. Regularization is used in the form of equalized learning rate by scaling each weight with $w_i' = w_i/c$, where c is the normalization constant per layer from He's initializer [10]. For the loss function, they employ the Wasserstein loss with regularization of R1 as mentioned in [11],[12].

StyleGAN introduced a new Generator architecture, employing a mapping network and a synthesis network as seen in *Figure* 1 [2]. The mapping network f is implemented using an 8-layer MLP (with dimensions 512×512 for each layer) and the synthesis network is a CNN. The goal of the mapping is to perform a non-linear mapping $f : Z \rightarrow W$ which translates the latent vector z to w.

StyleGAN introduced a new form of adaptive normalization, named $AdaIN$ (adaptive instance normalization) :

$$AdaIN(x_i, y) = y_{s,i}\frac{x_i - \mu(x_i)}{\sigma(x_i)} + y_{b,i} \quad (2)$$

where the input feature map is normalized with instance normalization first (μ and σ are the mean and standard deviation of the input feature map x_i, respectively, and a style is applied with $scale = y_s$ and $bias = y_b$.) In each layer, each feature map x_i is normalized separately and then scaled and biased using the corresponding scalar components of style y. "Style" in [2], refers to the main attributes of the data such as pose and identity. For improvement, StyleGAN introduces noise into spatial data to create stochastic variation with a factor B. More detailed information can be found in [2].

Just like ProGAN, StyleGAN makes use of the effective Wasserstein loss function based on Wasserstein distance metric [11].

3. RESULTS AND EXPERIMENT ANALYSIS

Table 1. Clean FID/KID scores

model	FID 50K (clean)↓	KID 50K (clean)↓
ProGAN	360.367	0.516
StyleGAN	312.049	0.435

The dataset consisted of 3.8K human embryo images spread between 10 classes with imbalances. Each class is

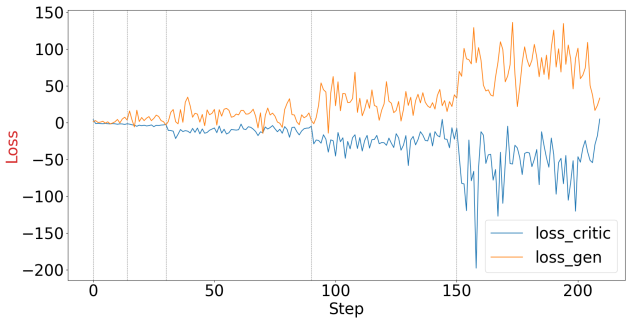
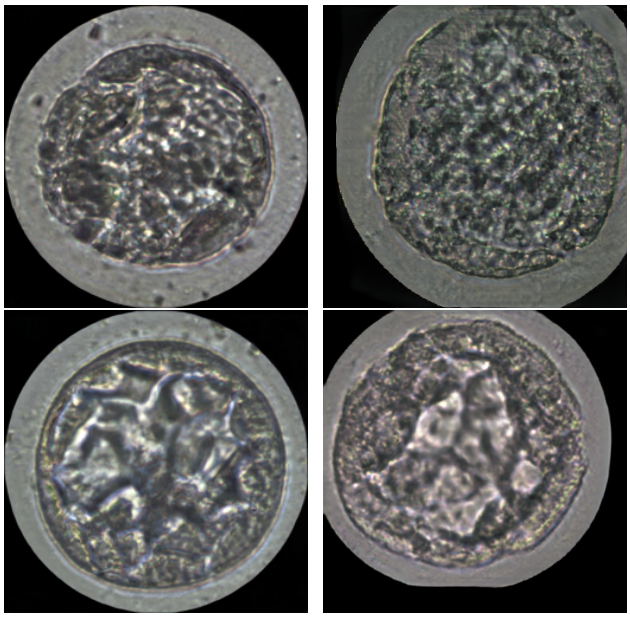


Fig. 2. ProGAN Loss in all resolutions (16×16 - 256×256).



(a) StyleGAN samples (b) ProGAN samples

Fig. 3. Generated images from trained instances of ProGAN and StyleGAN.

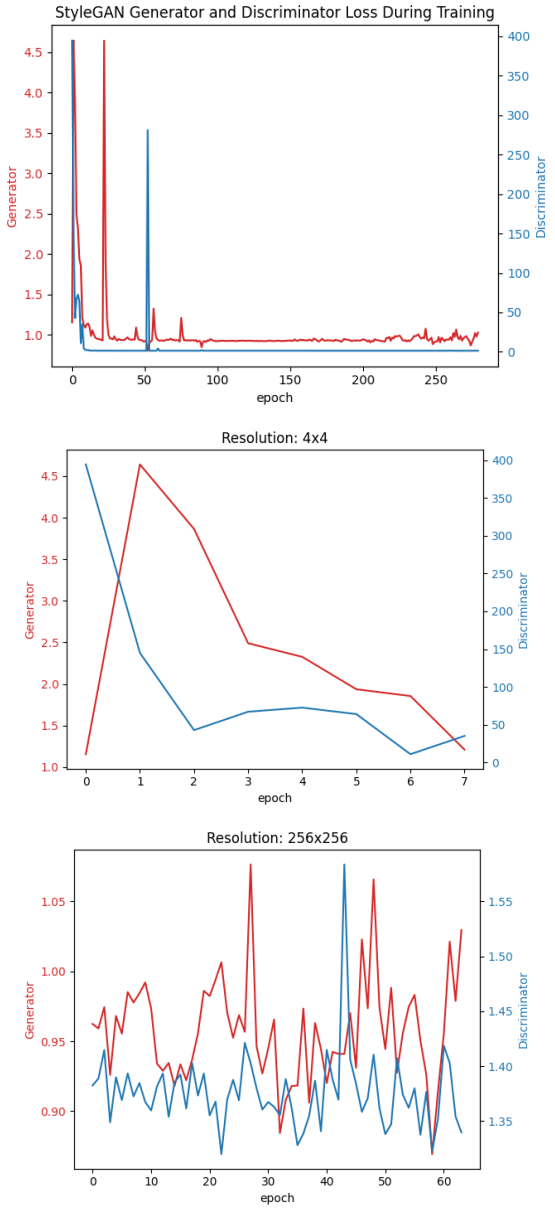


Fig. 4. Loss plots in different resolutions for StyleGAN. a) all resolutions, b) 8×8, c) 256×256

represented by two letters (AA, AB...etc.), with the first letter showing the inner cell mass score (ICM) and the second letter showing the Trophectoderm score (TE). The quality for each score goes as A,B...etc. in descending order. Results for ProGAN and StyleGAN can be observed below. StyleGAN managed to achieve better training stability and higher sample quality.

The synthetic image resolution for both GANs is 256×256, while the dataset on which the training was performed was resized accordingly. All networks were trained with *Adam* optimizer, a widely used optimizer for GAN training [13].

Adam optimizer was used with $\beta 1, \beta 2 = 0.0, 0.99$ and $\epsilon = 10^{-8}$ for both implementations with a fixed learning rate $lr = 0.001$.

For ProGAN the training progress started from 16×16 images and went up to 256×256. We declare that no image enhancements are applied to the dataset prior to training. Loss curves present an unusual spike in the resolutions of $> 128 \times 128$. Although regularization is used and step size increment is analogous to resolution, networks tend to overfit. Notable loss peaks are spotted at the beginning of the added layer training, probably because of a poor transition process.

StyleGAN had the following benefits: limited spike in loss curves, fast convergence (almost < 3 epochs), and partial independence from hyperparameters. The latter does not

mean that we can modify the model without consequences, rather than some aspects of the parameters are now less important such as: $\beta 1$, $\beta 2$ parameters of optimizers, the loss function, and the initial normalization of the model (it can even be removed).

Linear structure with 8-layered mapping network was used for the generator, with Linear Fully Connected neurons of $\{512, 512\}$ channels each. The batch sizes were configurable from $batch = 64$ for 4x4 resolution to $batch = 4$ for 256×256 resolution.

We report two computation metrics, measuring fidelity and similarity, the clean Frechet Inception Distance (FID) and the clean Kernel Inception Distance (KID)[14]. The lower the result, the better the performance.

4. CONCLUSION

We implemented two different GAN architectures, ProGAN and StyleGAN, to generate the human blastocyst image. The generated samples are evaluated based on sample quality, clean-FID, clean-KID, and loss functions. StyleGAN exceeds ProGAN in terms of image quality, performance metrics, and training stability, producing more stable loss curves. The models are suitable to create an augmented dataset, in order to be applied in other ML tasks.

5. COMPLIANCE WITH ETHICAL STANDARDS

This is a numerical simulation study for which *no* ethical approval was required.

6. ACKNOWLEDGMENTS

This research was carried out as part of the project "Classification and characterization of fetal images for assisted reproduction using artificial intelligence and computer vision" (Project code: KP6-0079459) under the framework of the Action "Investment Plans of Innovation" of the Operational Program "Central Macedonia 2014 –2020", that is co-funded by the European Regional Development Fund and Greece.

7. REFERENCES

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[2] Tero Karras, Samuli Laine, and Timo Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 4401–4410.

[3] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[4] Diederik P. Kingma and Max Welling, "An introduction to variational autoencoders," *Foundations and Trends® in Machine Learning*, vol. 12, no. 4, pp. 307–392, 2019.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[6] Darius Dirvanauskas, Rytis Maskeliūnas, Vidas Raudonis, Robertas Damaševičius, and Rafal Scherer, "Hemigen: human embryo image generator based on generative adversarial networks," *Sensors*, vol. 19, no. 16, pp. 3578, 2019.

[7] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[8] Angona Biswas, MD Nasim, Al Imran, Anika Tabassum Sejuty, Fabliha Fairooz, Sai Puppala, and Sajedul Talukder, "Generative adversarial networks for data augmentation," *arXiv preprint arXiv:2306.02019*, 2023.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[11] Martin Arjovsky, Soumith Chintala, and Léon Bottou, "Wasserstein gan," 2017, cite arxiv:1701.07875.

[12] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville, "Improved training of wasserstein gans," *Advances in neural information processing systems*, vol. 30, 2017.

[13] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[14] Gaurav Parmar, Richard Zhang, and Jun-Yan Zhu, "On aliased resizing and surprising subtleties in gan evaluation," in *CVPR*, 2022.