sensors

MDPI

*Article*

# HEMIGEN: Human Embryo Image Generator Based on Generative Adversarial Networks

**Darius Dirvanauskas** [1], **Rytis Maskeliūnas** [1], **Vidas Raudonis** [2], **Robertas Damaševičius** [3,4,*] and **Rafal Scherer** [5]

[1] Department of Multimedia Engineering, Kaunas University of Technology, 51368 Kaunas, Lithuania
[2] Department of Control Systems, Kaunas University of Technology, 51367 Kaunas, Lithuania
[3] Department of Software Engineering, Kaunas University of Technology, 51368 Kaunas, Lithuania
[4] Institute of Mathematics, Silesian University of Technology, 44-100 Gliwice, Poland
[5] Institute of Computational Intelligence, Czestochowa University of Technology, 42-200 Czestochowa, Poland
* Correspondence: robertas.damasevicius@ktu.lt

Received: 26 July 2019; Accepted: 14 August 2019; Published: 16 August 2019

**Abstract:** We propose a method for generating the synthetic images of human embryo cells that could later be used for classification, analysis, and training, thus resulting in the creation of new synthetic image datasets for research areas lacking real-world data. Our focus was not only to generate the generic image of a cell such, but to make sure that it has all necessary attributes of a real cell image to provide a fully realistic synthetic version. We use human embryo images obtained during cell development processes for training a deep neural network (DNN). The proposed algorithm used generative adversarial network (GAN) to generate one-, two-, and four-cell stage images. We achieved a misclassification rate of 12.3% for the generated images, while the expert evaluation showed the true recognition rate (TRR) of 80.00% (for four-cell images), 86.8% (for two-cell images), and 96.2% (for one-cell images). Texture-based comparison using the Haralick features showed that there is no statistically (using the Student's t-test) significant ($p < 0.01$) differences between the real and synthetic embryo images except for the sum of variance (for one-cell and four-cell images), and variance and sum of average (for two-cell images) features. The obtained synthetic images can be later adapted to facilitate the development, training, and evaluation of new algorithms for embryo image processing tasks.

**Keywords:** deep learning; neural network; generative adversarial network; synthetic images

## 1. Introduction

Deep neural networks (DNN) have become one of the most popular modern tools for image analysis and classification [1]. One of the first accurate implementations of DNN, AlexNet [2], was quickly bested by a deep convolutional activation feature (DeCAF) network, which extracted features from AlexNet and evaluated the efficacy of these features on generic vision tasks [3]. VGGNet demonstrated that increasing the depth of convolutional neural network (CNN) is beneficial for the classification accuracy [4]. However, the deeper neural network is more difficult to train. ResNet presented a residual learning framework to ease the training for very deep networks [5]. The ResNet architecture, which combined semantic information from a deep, coarse layer with appearance information from a shallow network, was adapted for image segmentation [6]. Region CNN (R-CNN) was proposed as a combination of high-capacity CNNs with bottom-up region proposals in order to localize and segment images [7].

The breakthrough of DNNs in quality led to numerous adaptations in solving medical image processing problems such as the analysis of cancer cells [8] and cancer type analysis [9]. Deep max-pooling CNNs were used to detect mitosis in breast histology images using a patch centered on the pixel as context [10]. The U-Net architecture with data augmentation and elastic deformations achieved very good performance on different biomedical segmentation applications [11]. A supervised max-pooling CNN was trained to detect cell pixels in regions that are preselected by a support vector machine (SVM) classifier [12]. After a pre-processing step to remove artefacts from the input images, fully CNNs were used to produce the embryo inner cell mass segmentation [13]. A set of Levenberg–Marquardt NNs trained using textural descriptors allowed for predicting the quality of embryos [14].

Image analysis methods were also applied to embryo image analysis. Techniques for extracting, classifying, and grouping properties were used to measure the quality of embryos. Real-time grading techniques for determining the number of embryonic cells from time-lapse microscope images help embryologists to monitor the dividing cells [15]. Conditional random field (CRF) models [16] were used for cell counting and assessing various aspects of the developing embryo and predicting the stage of the embryonic development. In addition to grading, embryonic positioning was achieved using linear chain Markov model [17]. Different solutions were developed for segmenting and calculating the number of cells: ImageJ, MIPAV, VisSeg [18]. Moreover, cell segmentation by marking its center and edges makes it possible to determine their shapes and quantities more quickly and accurately [19]. Two-stage classifier for embryo image classification was proposed in [20].

The rising progress of generative adversarial networks (GANs) applications on medical imaging state that most of the research is focused on synthetic imaging, the reconstruction, segmentation, and classification and proves the importance of the sector. Wasserstein-based GANs were applied for the synthesis of cells imaged by fluorescence microscopy capturing these relationships to be relevant for biological application [21]. The quality of generated artificial images also improved due to the improvements to deep neural networks [22] as well as on progressive growing of GANs for image data augmentation [23]. Variational autoencoders (VAE) and generative adversarial networks (GANs) are currently the most distinguished according to the quality of their results. The VAE models are more often used for image compression and recovery. Both methods have drawbacks, as recovering a part of the information loses some data and often introduces image fading or blurring. This effect could be reduced by matching the data as well as the loss distributions of the real and fake images by a pair of autoencoders used as the generator and the discriminator in the adversarial training [24]. Utilization of both generator and discriminator growing progressively from a low-resolution standpoint and adding new layers for an increase in fine details allowed for achieving a current state-of-the-art inception score of 8.80 on CIFAR10 dataset [25]. Network modifications can be applied to minimize fading of the resulting images [24]. Generative stochastic networks (GSN) were used to learn the transition operator of a Markov chain whose stationary distribution estimates the data distribution [26]. Deep recurrent attentive writer (DRAW) neural network architecture [27] was used to generate highly realistic natural images such as photographs of house numbers or other digits in the MNIST database. Dosovitskiy et al. [28] introduced an algorithm to find relevant information from the existing 3D chair models and to generate new chair images using that information. Denton et al. [29] introduced a generative parametric model capable of producing high-quality samples of natural images based on a cascade of convolutional networks within a Laplacian pyramid framework. Radford and Metz [30] introduced a simplification of training by a modification called deep convolutional GANs (DCGANs). GANs can also recover images from bridging text and image modelling, thus translating visual concepts from characters to pixels [31]. Zhang et al. [32] proposed stacked GANs (StackGAN) with conditioning augmentation for synthesizing photo-realistic images. GANs could also be applied for generating motion images, e.g., the motion and content decomposed GAN (MoCoGAN) framework for video generation by mapping a sequence of random vectors to a sequence of video frames [33]. The GAN for video (VGAN) model was based on a GAN with a spatial-temporal convolutional architecture that untangles the scene's foreground from the background and can be used at predicting plausible futures of static images [34]. Temporal GAN (TGAN) was used to learn a semantic representation of unlabeled videos by using different types of generators via Wasserstein GAN, and a method to train it stably in an end-to-end manner [35]. MIT presented a 3D generative adversarial network (3D-GAN) model to recreate 3D objects from a probabilistic space by leveraging recent advances in volumetric CNNs and GANs [36]. Li et al. [37] used multiscale GAN (DR-Net) to remove rain streaks from a single image. Zhu [38] used GANs for generating synthetic saliency maps for given natural images. Ma et al. [39] used background augmentation GANs (BAGANs) for synthesizing background images for augmented reality (AR) applications. Han et al. presented a system based on a Wasserstein GANs for generating realistic synthetic brain MR images [40].

All these results are achieved by using large expert-annotated and ready-made databases and also exposing the problem of lacking good, core training datasets. In this work, we aimed to develop a method to generate realistic synthetic images that could later be used for classification, analysis, and training, thus resulting in the creation of novel synthetic datasets for research areas lacking data such as human embryo images. Here we used human embryo images obtained during cell development processes for training a DNN. We propose an algorithm for generating one-, two-, and four-cell images (the selection was based on the initial dataset provided by our medical partners) to increase the overall number of unique images available for training. For generating images, we have developed a generative adversarial network for image recovery, filling, and improvement. The significance of our approach would be that the method was applied to embryonic cell images as a type of image. It was very important for the GAN to accurately reproduce the outline of the cell (often poorly visible even in microscopy picture), since the whole image was almost the same (gray) and the cell itself is translucent. Our focus was not only to generate the generic image of a cell as such, but to make sure that it has all necessary attributes of a real cell image to provide a fully realistic synthetic version. We believe that as a large number of real embryo images required for training neural networks are difficult to obtain due to ethics requirements, the synthetic images generated by the GAN can be later adapted to facilitate the development, training, and evaluation of new algorithms for embryo image-processing tasks.

## 2. Materials and Methods

### 2.1. Architecture of the Generative Adversarial Network (GAN)

The generative adversarial network (GAN) consists of two parts: A generator *G* and a discriminator *D* (Figure 1).

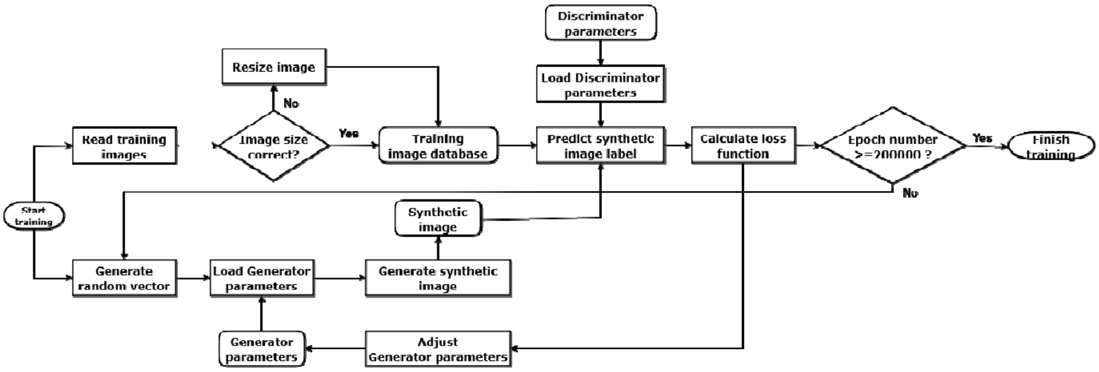


**Figure 1.** A pipeline of a generative adversarial network (GAN).

The discriminator tries to distinguish true images from synthetic, generator-generated images. The generator tries to create images with which the discriminator could be deceived. The discriminator is implemented as a fully connected neural network with dense layers that represents an image as a probability vector and classifies it into two classes, a real or a fake image. The generator is a reverse model that restores the former image from a random noise vector. During the training, the discriminator is trained to maximize the probability of assigning the correct class to the training images and images generated by the generator. At the same time, the generator is trained to minimize classification error $\log(1 - D(G(z)))$. This process can be expressed in terms of game theory as a two-player (discriminator and generator) minimax game with cost function $V(G, D)$ [41]:

$$\max_G \min_D V(D, G) = E_{x \sim P(x)}[\log(D(x))] + E_{z \sim P(z)}[\log(1 - D(G(z)))]. \tag{1}$$

The discriminator evaluates the images created by the training database and the generator. The architecture of the discriminator network is shown in Figure 2. The network consists of six layers. The monochrome 200 × 200 pixel image in the first layer is transformed and expanded into a single line vector. The dense layer is used in the second, fourth, and sixth layers. The LeakyReLU [42] function, where $\alpha = 0.2$, is used in the third and fifth layers as it allows a small gradient when the unit is not active:

$$\text{LReLu}(x) = \max(x, 0) + \alpha \min(x, 0). \tag{2}$$

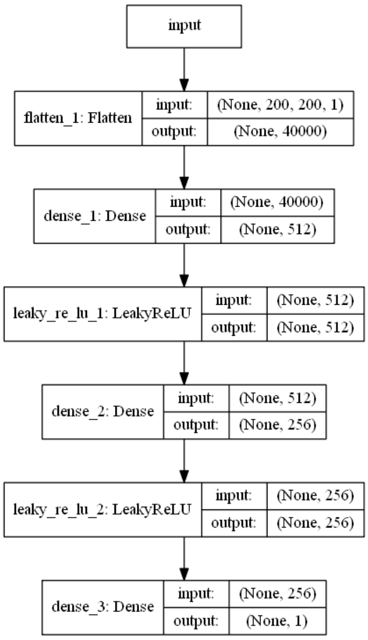


**Figure 2.** Architecture of the discriminator network.

At the network output, we get a one if the network guesses that the image is real or zero if the network guesses that the image is fake.

The generator is composed of 11 layers (Figure 3). The input layer contains 1 × 100 vector of randomly generated data. These data are transformed using the dense, LeakyReLU, and batch normalization layers. The ReLU layer uses $\alpha = 0.2$. The batch normalization layer was used to normalize the activations of the previous layer at each batch, by applying a transformation that maintains the mean activation close to 0 and the activation standard deviation close to 1 [43]:

$$\hat{x}_i \leftarrow \frac{x_i - \mu_\beta}{\sqrt{\sigma_\beta^2 + \varepsilon}}. \tag{3}$$

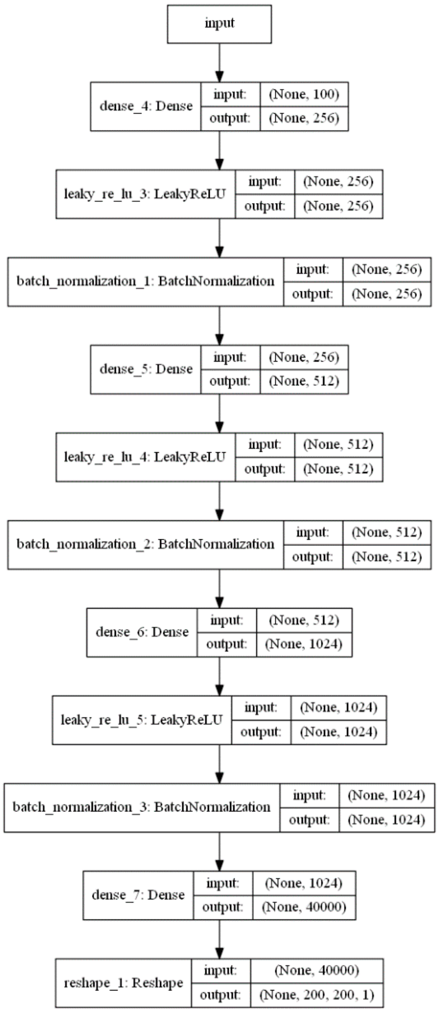


**Figure 3.** Architecture of the generator network.

The batch normalization momentum value was set to 0.8, while all other values were used as per default Keras model values. The output produces a 200 × 200 pixel sized black-and-white image.

## 2.2. Training

We trained one (single) network to generate all types of cell images. For the training, we have used the Adam optimization algorithm [44] with a fixed training speed of 0.00002, and $\beta_1 = 0.7$ for training the generator. During training, the Adam algorithm calculates the gradient as follows:

$$g_t \leftarrow \Delta_\theta f_t(\theta_{t-1}). \tag{4}$$

The biased first moment estimate is updated as follows:

$$m_t \leftarrow \beta_1 m_{t-1} + (1 - \beta_1) g_t. \tag{5}$$

The biased second moment estimate is update as follows:

$$v_t \leftarrow \beta_2 v_{t-1} + (1 - \beta_2) g_t^2. \tag{6}$$

Then we compute the bias corrected first moment estimate as follows:

$$\hat{m}_t \leftarrow \frac{m_t}{1-\beta_1^t}. \tag{7}$$

The bias corrected second moment estimate is computed as follows:

$$\hat{v}_t \leftarrow \frac{v_t}{1-\beta_2^t}. \tag{8}$$

This gives the update parameters as follows:

$$\theta_t \leftarrow \frac{\theta_{t-1} - a\hat{m}_t}{\sqrt{\hat{v}_t}'' + \varepsilon}. \tag{9}$$

The binary cross-entropy function is used to evaluate the error as follows:

$$-\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)]. \tag{10}$$

## 2.3. Evaluation of Image Quality

The problem with using the GAN model is that there are no clear ways to measure it qualitatively. There are few most commonly used criteria to evaluate GAN network: Average log-likelihood, classifier classification [41], and visual fidelity of samples [45]. These methods have both advantages and disadvantages. Log-likelihood and classifiers attribute the image to a certain class with a matching factor. However, they only indicate whether the generated image is like the average of one of the classes but does not value a qualitatively generated image. Such a qualitative assessment requires a human evaluator, but with a large amount of data, such estimates can change independently and be biased [46]. Likewise, a different expert person could evaluate the same data somewhat differently.

We use a combined method to evaluate the results obtained in our study, which includes: (1) Human expert evaluation, (2) histogram comparison, and (3) texture feature comparison. The generated images are evaluated by human experts (three medical research professionals) to determine if they are being reproduced qualitatively. If we can determine its class from the received image, the image is quality-restored. The expert based scoring was calculated using a visual Turing test [47] as this method has already been proven effective in the evaluation of synthetic images generated by GANs [48].

The histogram comparison method checks whether the histogram of the generated images corresponds to the histogram of the images in the training database. For comparison, we used the normalized average histogram of training data $H_1$, and the normalized average histogram of generated images $H_2$. Further, we apply four different methods for comparison as it is recommended in [49]. The correlation function determines the likelihood of two image histograms. Its value of two identical histograms equals one:

$$C(H_1, H_2) = \frac{\sum_I \left(H_1(I) - \overline{H}_1\right)\left(H_2(I) - \overline{H}_2\right)}{\sqrt{\sum_I \left(H_1(I) - \overline{H}_1\right)^2 \sum_I \left(H_2(I) - \overline{H}_2\right)^2}}. \tag{11}$$

The Chi-square function takes the squared difference between two histograms. The squared differences are divided by the number of samples, and the sum of these weighted squared differences is the likelihood value:

$$\chi^2(H_1, H_2) = \sum_I \frac{(H_1(I) - H_2(I))^2}{H_1(I)}. \tag{12}$$

The histogram intersection calculates the similarity of two discretized histograms, with a possible value of the intersection lying between no overlap and identical distributions. It works well on categorical data and deals with null values by making them part of the distribution.

$$I(H_1, H_2) = \sum_I \min(H_1(I), H_2(I)) \tag{13}$$

The Bhattacharyya distance approximates the normalized distance between the histograms using the maximum likelihood of two object histograms as follows:

$$B(H_1, H_2) = \sqrt{1 - \frac{1}{\sqrt{H_1 H_2 N^2}} \sum_I \sqrt{H_1(I) H_2(I)}.} \tag{14}$$

The texture-based comparison uses the texture analysis features based on grey level co-occurrence matrix (GLCM), which are related to second-order image statistics that were introduced by Haralick [50] as follows: (1) Angular second moment (energy), (2) contrast, (3) correlation, (4) variance, (5) inverse difference moment (homogeneity), (6) sum average, (7) sum variance, (8) sum entropy, (9) entropy, (10) difference variance, (11) difference entropy, (12) information measure of correlation, (13) information measure of correlation II, and (14) maximal correlation coefficient.

### *2.4. Ethics Declaration*

The study was conducted in accordance with the Declaration of Helsinki, and the protocol was approved by the Ethics Committee of Faculty of Informatics, Kaunas University of Technology (No. IFEP201809-1, 2018-09-07).

## 3. Experiment and Results

### *3.1. Dataset and Equipment*

We have used the embryo photos taken with Esco Global incubator series, called Miri TL (time lapse). The embryo image sets were registered in the German, Chinese, and Singapore clinics. No identity data were ever provided to the authors of this paper. The Esco's embryo database consists of three image classes that together have 5000 images with a resolution of 600 × 600 pixels. Number of different images in classes: One-cell—1764 images, two-cell—1938 images, four-cell—1298 images. Images are obtained from 22 different growing embryos in up to five stages of cell evolution. The database was then magnified by rotating each photo at 90, 180, and 270 degrees. Resulting number of different images in classes: (one-cell images—1764 × 4 = 7056; two-cell images—1938 × 4 = 7752; four-cell images—1298 × 4 = 5192). The images were taken in the following culturing conditions: Temperature of 37 °C, stable level of 5% CO2, and the controllable values of nitrogen and oxygen mixture. The embryos were photographed in a culture coin dish made from polypropylene with a neutral media pH = 7. The inverted microscopy principle was used, with 20x lenses, without zoom, with focusing and with field of view of 350 um. A camera sensor used was IDS UI-3260CP-M-GL (IDS Imaging Development Systems GmbH, Obersulm, Germany). An example of embryo images in the Esco dataset is presented in Figure 4. The evaluation and training of GANs was done on an Intel i5-4570 CPU with a GeForce 1060 GPU and 8 GB of RAM.

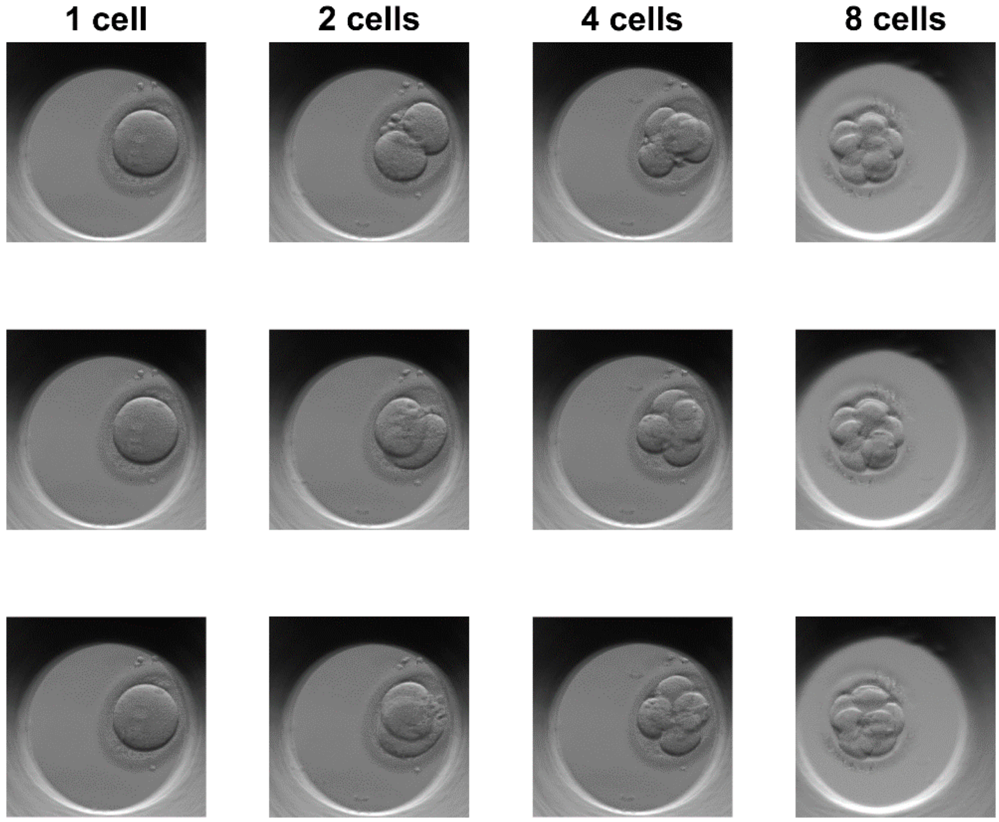


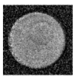
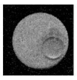
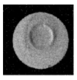
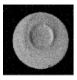
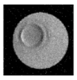
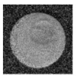
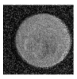
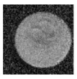
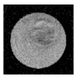
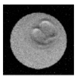
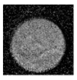
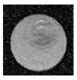
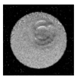
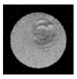
**Figure 4.** Sample images from Esco embryo image dataset.

### *3.2. Results*

The training of GAN was carried out in 200,000 iterations, with a sample size of 256 images per batch. The duration of training using GeForce 1060 GPU was about 15 h 45 min.

From the final generated images, one can easily count the number of embryonic cells and determine the class (Figure 5). During subsequent iterations, the generator restored the embryo image and reduced the amount of noise in the generated images.

**Table 1.** Evolution of embryo cell images during GAN training.

| **Epoch** | **0** | **25,000** | **50,000** | **75,000** | **100,000** | **125,000** | **150,000** | **175,000** | **200,000** |
|---|---|---|---|---|---|---|---|---|---|
| one-cell | |  | |  |  |  | | |  |
| two-cells | |  |  |  |  |  | | | |
| four-cells | |  |  |  | |  | | | |

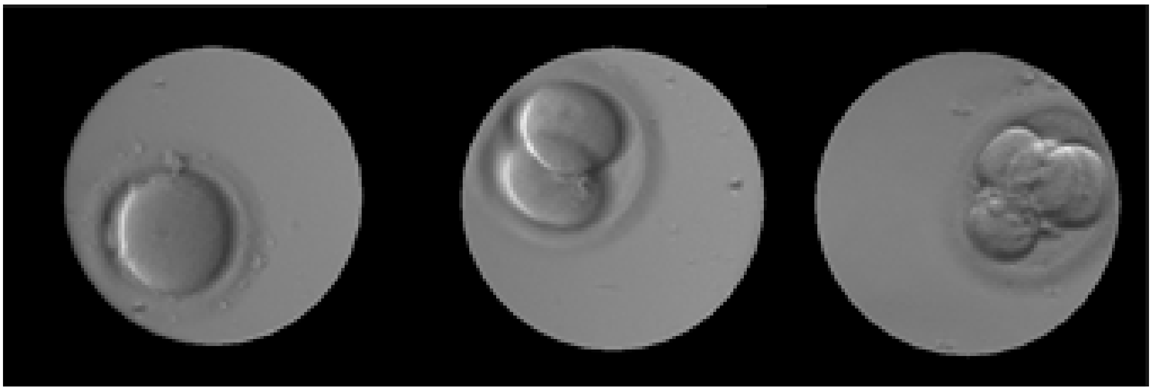


**Figure 5.** Example of final generated embryo cell images using the proposed algorithm. From left to right: One, two, and four cells. Images were filtered from "salt and pepper" noise using a median filter. See Table 1 for raw outputs.

During the training, embryo cell images were generated at every 25,000 iterations. An example of these images is shown in Table 1.

After 25,000 iterations, we can see that he managed to restore the plate's flair and release the background, but the embryo itself was not restored. After 100,000 iterations, the generator has been able to clearly reproduce the image of the embryo cell.

The error function values for both generator and discriminator are shown in Figures 6 and 7. During an initial training up to 20,000 iterations, the error function value of the generator increases rapidly. Further increase of the error function results in a slower pace. Once the training number reaches 75,000 iterations, the value of the generator error function begins to increase with the training of two-cell and four-cell images. The generator becomes able to produce more complex images, which is also visible from the generated sample images (see Table 1). The one-cell image can be identified from 50,000 iterations, whereas two-cell and four-cell images can be recognized from 75,000–125,000 iterations. From the discriminator error feature, we can see that starting from 20,000 iterations it becomes difficult to separate real images from artificially generated images, and the value of the error function is less than 0.15. A value of discriminator log loss increases, as discriminator is unable to differentiate between a real image and a generated image.

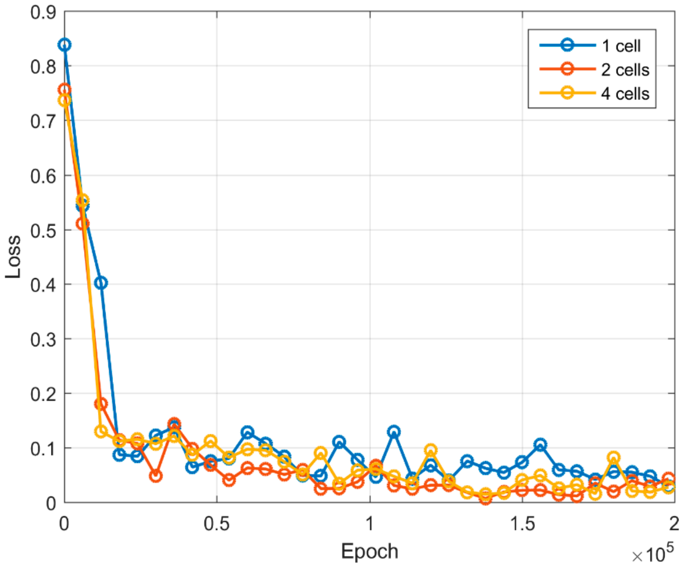


**Figure 6.** Loss log graph for the generator network.

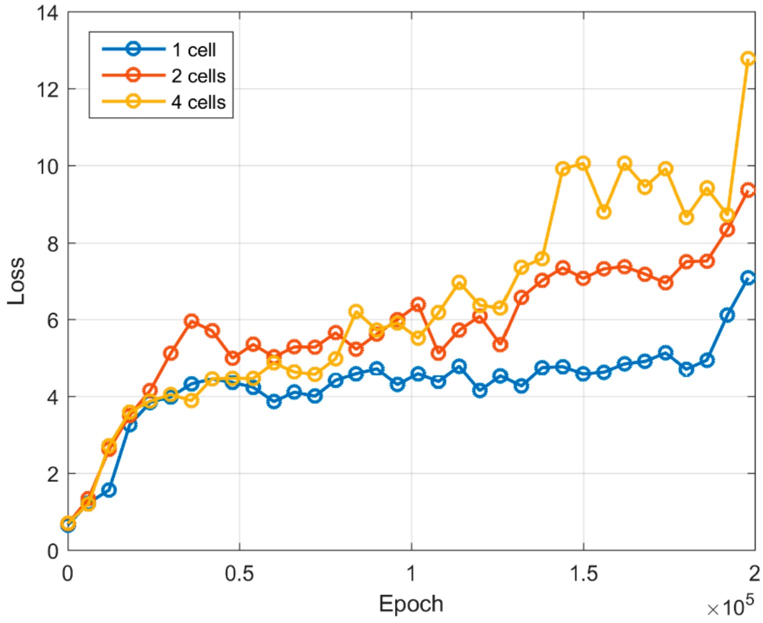


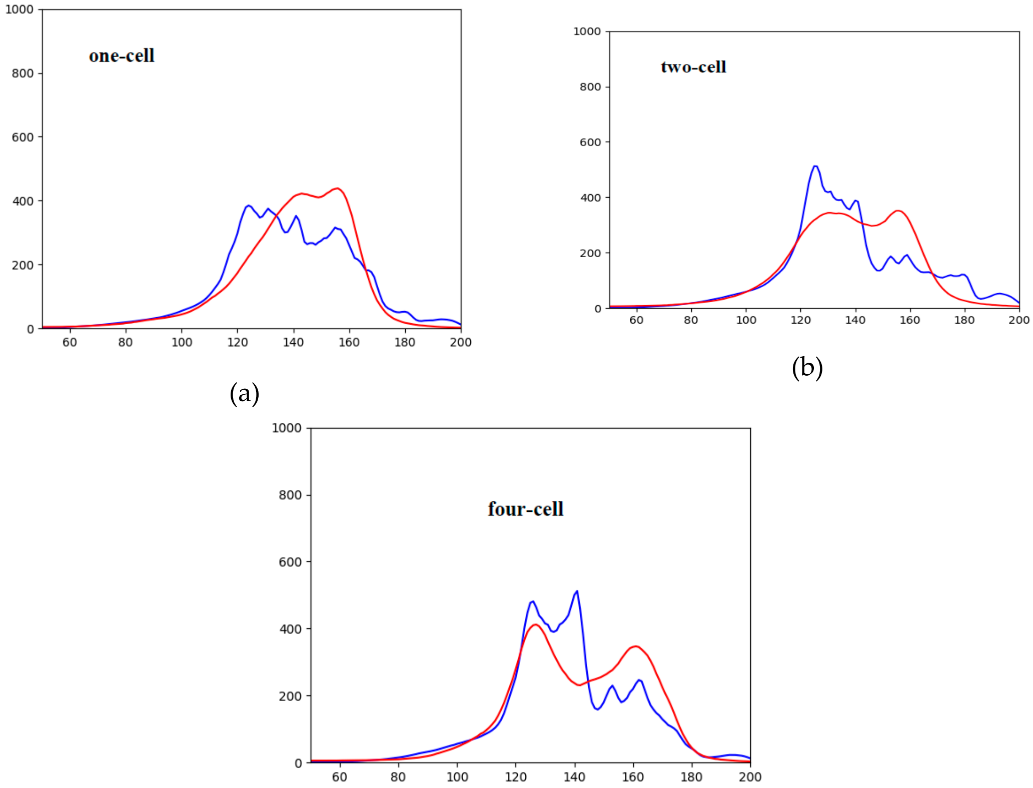
**Figure 7.** Loss log graph for the discriminator network.

For the expert-based evaluation (see Table 2), 1500 images (500 in each class) were generated. The number of artificial one-cell images, where one cell was clearly recognized, was 96.2%. The quality of two-cell and four-cell image generation of more complex images deteriorated. Of the two-cell images, the number of cells could be clearly determined from 86.8% of the images. When evaluating the four-cell images, 80% accuracy was obtained, i.e., one out of five images was generated inaccurately (as decided by an expert).

**Table 2.** Evaluation of the generated embryo image cells by human experts.

| Class | Test Images (Generated) | Good Images | Accuracy by Expert Selection |
|---|---|---|---|
| one-cell | 500 | 481 | 96.2% |
| two-cells | 500 | 434 | 86.8% |
| four-cells | 500 | 400 | 80.00% |

Additionally, the image similarity was evaluated by comparing their histograms. In Figure 8, one can see the comparison of image histograms, where the blue curve is the average histogram of the training image dataset and the red is the average histogram of the artificially generated images. In all three classes, we can see that the generated image is brighter than the real one. This could be explained due to the slight "salt-and-pepper"-type noise in the generated image. The highest histogram match was obtained in the single-cell-generated images. This shows that the generator was better able to reproduce images of a simpler structure. Note that Figure 8 shows an evaluation of a generated cell image with a maximum epoch number of 200,000. The particular histogram cannot be identical as a single network generates images of different cells. This comparison was done by evaluating the average histogram of all (training and generated) image histograms.



(a)

(b)

(c)

**Figure 8.** Comparison of histograms of real vs. generated images (real images—blue, generated images—red): (**a**) one-cell images, (**b**) two-cell images, (**c**) four-cell images.

Table 3 shows the results of a comparison of normalized histograms using correlation, Chi-square, intersection, and Bhattacharyya distance formulas. In all four cases, the largest histogram coincidence was shown by the one-cell generated images, while the two-cell and four-cell images performed worse.

**Table 3.** Evaluation of generated artificial embryo cell images using histogram comparison criteria.

| | **One-cell** | **Two-cells** | **Four-cells** |
|---|---|---|---|
| Correlation | 0.995 | 0.990 | 0.986 |
| Chi-square | 0.236 | 0.455 | 0.442 |
| Intersection | 1.883 | 1.849 | 1.873 |
| Bhattacharyya | 0.147 | 0.210 | 0.208 |

We also compared the values of the textural (Haralick) features for the original and generated embryo images and evaluated the statistical difference between the values of features using the Student's t-test. The results were significant ($p < 0.01$) for all Haralick features except the sum variance feature (one-cell images), variance and sum average (two-cell images), and the sum variance feature (four-cell images).

To compare the Haralick features from both (original and generated) image datasets, we also used the Pearson correlation between the principal components of feature sets extracted using principal component analysis (PCA). For both feature sets, the first principal component (PC1) explains more than 98% of variance, so we use only PC1 for further comparison. The results are presented in Table 4 and show a very high correlation between the PC vectors.

**Table 4.** Results of principal component analysis (PCA) of Haralick features from original and synthetic embryo images.

| | **One-cell** | **Two-cells** | **Four-cells** |
|---|---|---|---|
| Explained variance of PC1 (original) | 99.90% | 99.94% | 99.86% |
| Explained variance of PC1 (synthetic) | 98.95% | 99.60% | 99.73% |
| Correlation between values of PC1 (original) and PC1 (synthetic) | 1.0000 | 0.9997 | 0.9999 |

## 4. Discussion

As the correct comparison with other algorithms not possible due to very different source datasets used, we tried to compare within the misclassification rate, by comparing percentage of the images from all generated groups were assigned to correct class or not at all. Our HEMIGEN method was rated at 12.3%.

Table 5 provides a comparison of the misclassification rate of synthetic images when compared to the results obtained by other authors. Gaussian mixture deep generative network (DGN) demonstrated 36.02% misclassification rate [51]. DGN with auxiliary variables and two stochastic layers and skip connections achieved the 16.61% misclassification rate [52]. Semi-supervised classification and image generation with four-layer generator demonstrated 8.11% misclassification rate on house number image generation [48]. The adversarial learned inference (ALI) model, which jointly learns a generation network and an inference network using an adversarial process, reached a misclassification rate of 7.42% using CIFAR10 test set (tiny images) [53]. The WGAN-based approaches ranged from 6.9% [21] to 50% [40] depending on the application. The methods indicated are only “loosely” comparable, taking into account the differences (importance) of features of synthetic image targeted and scopes of the works by other researchers.

**Table 5.** Misclassification rate.

| **Model** | **Images Analyzed** | **Overall Misclassification Rate** |
|---|---|---|
| DGN [51] | Digits 0 to 9 and combination (based on SVHN, and NORB sets) (*realistic images*) | 36.02% |
| Skip Deep Generative Model [52] | Digits 0 to 9 and combination (based on SVHN, and NORB sets) (*realistic images*) | 16.61% |
| GAN (feature matching) [48] | LSVRC2012 dataset with 1,000 categories (*non-realistic images*) | 8.11% |
| ALI [53] | 25 types of shapes (*non-realistic images*) | 7.42% |
| WGAN [40] | 6 classes of brain MR images (*realistic images*) | 50% |
| WGAN-GP [21] | 8 classes of florescent microscopy images (*realistic images*) | 6.9% |
| HEMIGEN (our approach) | Once cell, Two cells, Four cells (*realistic images*) | 12.3% |

We also checked for the problem of mode collapse where the generator would produce the same image over and over, thus fooling discriminator, while no new image would be generated in real-life. We have not found this case in our approach, possibly due to an adequate number of different embryos used. Some generated images had embryos, which overlap one another, but this was not considered a failure, but a realistic case of cell division.

## 5. Conclusions

We used generative adversarial networks trained on real human embryo cell images to generate a dataset of synthetic one-, two-, and four-cell embryo images. We have achieved the highest quality of generated images for single-cell embryo images, where 96.2% of the synthetic embryo images were recognized as accurate and usable by human experts. The worst accuracy was achieved for the synthetic four-cell images, of which only 80% could be identified correctly. These results were confirmed by the histogram comparison, which achieved the highest scores for synthetic single-cell images (an average correlation of 0.995 was achieved when comparing histograms of real and synthetic one-cell embryo images), as well as by comparison of image textures analyzed using the Haralick features.

As our algorithm allows us to manipulate the size, position, and number of the artificially generated embryo cell images, these images can then be used to train and validate other embryo image processing algorithms, when the real embryo images are not available, or the number of available real embryo images is too small for training neural networks.

**Author Contributions:** Conceptualization, R.M. and V.R.; data curation, D.D.; funding acquisition, R.S.; investigation, D.D, R.M., and V.R.; methodology, R.M.; software, D.D; supervision, R.M.; validation, D.D and R.M.; visualization, D.D; writing—original draft, D.D, R.M., and R.D.; writing—review and editing, R.M. and R.S.

**Funding:** The project partially financed under the program of the Polish Minister of Science and Higher Education under the name "Regional Initiative of Excellence" in the years 2019–2022 project number 020/RID/2018/19, the amount of financing 12,000,000.00 PLN.

**Acknowledgments:** The authors also would like to thank Esco Global for kindly provided embryo image dataset.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211. [CrossRef]
2. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
3. Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; Darrell, T. DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. In Proceedings of the 31st International Conference on Machine Learning, Beijing, China, 21–26 June 2014; Volume 32, pp. 647–655.
4. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Proceedings of the International Conference on Learning Representations ICLR, San Diego, CA, USA, 7–9 May 2015; pp. 1–14.
5. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jersey, NJ, USA, 26 June–1 July 2016; IEEE: Piscataway, NJ, USA. [CrossRef]
6. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. [CrossRef]
7. Girshick, R.B.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '14), Washington, DC, USA, 23–28 June 2014; pp. 580–587. [CrossRef]
8. Esteva, A.; Kuprel, B.; Novoa, R.A.; Ko, J.; Swetter, S.M.; Blau, H.M.; Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature* **2017**, *542*, 115–118. [CrossRef] [PubMed]
9. Fakoor, R.; Ladhak, F.; Nazi, A.; Huber, M. Using deep learning to enhance cancer diagnosis and classification. In *ICML Workshop on the Role of Machine Learning in Transforming Healthcare Proceedings of the International Conference on Machine Learning, Atlanta, AG, USA, 16–21 June 2013*; ACM: New York, NY, USA, 2013; pp. 1–7.
10. Ciresan, D.C.; Giusti, A.; Gambardella, L.M.; Schmidhuber, J. Mitosis Detection in Breast Cancer Histology Images with Deep Neural Networks. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention, MICCAI 2013, Nagoya, Japan, 22–26 September 2013; pp. 411–418.
11. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the Medical Image Computing and Computer-Assisted Intervention, MICCAI 2015, Munich, Germany, 5–9 October 2015; Lecture Notes in Computer Science. Springer: Cham, Switzerland, 2015; Volume 9351, pp. 234–241.
12. Dong, B.; Shao, L.; Da Costa, M.; Bandmann, O.; Frangi, A.F. Deep learning for automatic cell detection in wide-field microscopy zebrafish images. In Proceedings of the 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), Brooklyn, NY, USA, 16–19 April 2015; pp. 772–776.

13. Kheradmand, S.; Singh, A.; Saeedi, P.; Au, J.; Havelock, J. Inner cell mass segmentation in human HMC embryo images using fully convolutional network. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 1752–1756.
14. Manna, C.; Nanni, L.; Lumini, A.; Pappalardo, S. Artificial intelligence techniques for embryo and oocyte classification. *Reprod. Biomed. Online* **2013**, *26*, 42–49. [CrossRef] [PubMed]
15. Wang, Y.; Moussavi, F.; Lorenzen, P. Automated Embryo Stage Classification in Time-Lapse Microscopy Video of Early Human Embryo Development. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI (2013), Nagoya, Japan, 22–26 September 2013; pp. 460–467. [CrossRef]
16. Khan, A.; Gould, S.; Salzmann, M. Automated monitoring of human embryonic cells up to the 5-cell stage in time-lapse microscopy images. In Proceedings of the 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), New York, NY, USA, 16–19 April 2015; pp. 389–393.
17. Khan, A.; Gould, S.; Salzmann, M. A Linear Chain Markov Model for Detection and Localization of Cells in Early Stage Embryo Development. In Proceedings of the 2015 IEEE Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 6–9 January 2015; pp. 526–533.
18. Nandy, K.; Kim, J.; McCullough, D.P.; McAuliffe, M.; Meaburn, K.J.; Yamaguchi, T.P.; Gudla, P.R.; Lockett, S.J. Segmentation and Quantitative Analysis of Individual Cells in Developmental Tissues. In *Mouse Molecular Embryology; Methods in Molecular Biology (Methods and Protocols)*; Lewandoski, M., Ed.; Humana Press: Boston, MA, USA, 2014; Volume 1092, pp. 235–253.
19. Baggett, D.; Nakaya, M.-A.; McAuliffe, M.; Yamaguchi, T.P.; Lockett, S. Whole cell segmentation in solid tissue sections. *Cytometry* **2005**, *67A*, 137–143. [CrossRef] [PubMed]
20. Dirvanauskas, D.; Maskeliunas, R.; Raudonis, V.; Damasevicius, R. Embryo development stage prediction algorithm for automated time lapse incubators. *Comput. Methods Programs Biomed.* **2019**, *177*, 161–174. [CrossRef] [PubMed]
21. Osokin, A.; Chessel, A.; Salas, R.E.C.; Vaggi, F. GANs for Biological Image Synthesis. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; IEEE: Piscataway, NJ, USA. [CrossRef]
22. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive growing of GANs for improved quality, stability, and variation. *arXiv* **2017**, arXiv:1710.10196.
23. Han, C.; Rundo, L.; Araki, R.; Furukawa, Y.; Mauri, G.; Nakayama, H.; Hayashi, H. Infinite Brain MR Images: PGGAN-based Data Augmentation for Tumor Detection. *arXiv* **2019**, arXiv:1903.12564.
24. Khan, S.H.; Hayat, M.; Barnes, N. Adversarial Training of Variational Auto-encoders for High Fidelity Image Generation. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Waikoloa Village, HI, USA, 12–15 March 2018; IEEE: Piscataway, NJ, USA, 2018. [CrossRef]
25. Yi, X.; Walia, E.; Babyn, P. Generative adversarial network in medical imaging: A review. *arXiv* **2018**, arXiv:1809.07294.
26. Bengio, Y.; Thibodeau-Laufer, É.; Alain, G.; Yosinski, J. Deep Generative Stochastic Networks Trainable by Backprop. In Proceedings of the 31st International Conference on International Conference on Machine Learning, Beijing, China, 21–26 June 2014; Volume 32, pp. II-226–II-234.
27. Gregor, K.; Danihelka, I.; Graves, A.; Jimenez Rezende, D.; Wierstra, D. DRAW: A Recurrent Neural Network for Image Generation. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; Volume 37, pp. 1462–1471.
28. Dosovitskiy, A.; Springenberg, J.T.; Brox, T. Learning to generate chairs with convolutional neural networks. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 1538–1546.
29. Denton, E.; Chintala, S.; Szlam, A.; Fergus, R. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 1 (NIPS'15), Montreal, QC, Canada, 7–12 December 2015; MIT Press: Cambridge, MA, USA, 2015; Volume 1, pp. 1486–1494.
30. Radford, A.; Metz, L.; Chintala, S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *arXiv* **2015**, arXiv:1511.06434.

31. Reed, S.; Akata, Z.; Yan, X.; Logeswaran, L.; Schiele, B.; Lee, H. Generative Adversarial Text to Image Synthesis. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; Volume 48, pp. 1060–1069.
32. Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; Metaxas, D. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5908–5916.
33. Tulyakov, S.; Liu, M.-Y.; Yang, X.; Kautz, J. MoCoGAN: Decomposing Motion and Content for Video Generation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake, UT, USA, 18–22 June 2018; pp. 1526–1535.
34. Vondrick, C.; Pirsiavash, H.; Torralba, A. Generating Videos with Scene Dynamics. In Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16), Long Beach, CA, USA, 4–9 December 2017; pp. 613–621.
35. Saito, M.; Matsumoto, E.; Saito, S. Temporal Generative Adversarial Nets with Singular Value Clipping. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2849–2858. [CrossRef]
36. Wu, J.; Zhang, C.; Xue, T.; Freeman, W.T.; Tenenbaum, J.B. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling. In Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16), Boston, MA, USA, 15–19 September 2016; pp. 82–90.
37. Li, C.; Guo, Y.; Liu, Q.; Liu, X. DR-Net: A Novel Generative Adversarial Network for Single Image Deraining. *Secur. Commun. Netw.* **2018**, *2018*, 7350324. [CrossRef]
38. Zhu, D.; Dai, L.; Luo, Y.; Zhang, G.; Shao, X.; Itti, L.; Lu, J. Multi-Scale Adversarial Feature Learning for Saliency Detection. *Symmetry* **2018**, *10*, 457. [CrossRef]
39. Ma, Y.; Liu, K.; Guan, Z.; Xu, X.; Qian, X.; Bao, H. Background Augmentation Generative Adversarial Networks (BAGANs): Effective Data Generation Based on GAN-Augmented 3D Synthesizing. *Symmetry* **2018**, *10*, 734. [CrossRef]
40. Han, C.; Hayashi, H.; Rundo, L.; Araki, R.; Shimoda, W.; Muramatsu, S.; Nakayama, H. GAN-based synthetic brain MR image generation. In Proceedings of the 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), Washington, DC, USA, 4–7 April 2018; IEEE: Piscataway, NJ, USA, 2018. [CrossRef]
41. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. In Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2 (NIPS'14), Montreal, QC, Canada, 8–13 December 2014; MIT Press: Cambridge, MA, USA, 2014; pp. 2672–2680.
42. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic model. In Proceedings of the International Conference on Machine ICML 2013, Atlanta, GA, USA, 16–21 June 2013; Volume 30, p. 3.
43. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on International Conference on Machine Learning-Volume 37 (ICML'15), Lille, France, 7–9 July 2015; Volume 37, pp. 448–456.
44. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations, Indianapolis, IN, USA, 24–28 March 2014.
45. Theis, L.; van den Oord, A.; Bethge, M. A note on the evaluation of generative models. In Proceedings of the International Conference on Learning Representations (ICLR), San Juan, Puerto Rico, 19 November 2015.
46. Im, D.J.; Kim, C.D.; Jiang, H.; Memisevic, R. Generating Images with Recurrent Adversarial Networks. *arXiv* **2016**, arXiv:1602.05110.
47. Geman, D.; Geman, S.; Hallonquist, N.; Younes, L. Visual Turing test for computer vision systems. In Proceedings of the National Academy of Sciences, Washington, DC, USA, 22–29 December 2015; p. 201422953. [CrossRef]
48. Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training GANs. In Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16), Barcelona, Spain, 5–10 December 2016; Curran Associates Inc.: Red Hook, NY, USA, 2016; pp. 2234–2242.
49. Tazehkandi, A.A. *Computer Vision with OpenCV 3 and Qt5*; Packt Publishing: Birmingham, UK, 2018.

50. Haralick, R.M.; Shanmugam, K.; Dinstein, I. Textural Features for Image Classification. *IEEE Trans. Syst. Man Cybern.* **1973**, *SMC-3*, 610–621. [CrossRef]
51. Kingma, D.P.; Rezende, D.J.; Mohamed, S.; Welling, M. Semi-supervised learning with deep generative models. In Proceedings of the 27th International Conference on Neural Information Processing Systems-Volume 2 (NIPS'14), Montreal, QC, Canada, 8–13 December 2014; MIT Press: Cambridge, MA, USA, 2014; pp. 3581–3589.
52. Maaløe, L.; Sønderby, C.K.; Sønderby, S.K.; Winther, O. Auxiliary deep generative models. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 19–24 June 2016; Volume 48, pp. 1445–1453.
53. Dumoulin, V.; Belghazi, I.; Poole, B.; Lamb, M.A.; Mastropietro, O.; Courville, A. Adversarially Learned Inference. In Proceedings of the International Conference on Learning Representations (ICLR), Toulon, France, 24–26 April 2017.