IMAGE CLASSIFICATION WITH GAN-BASED DATA AUGMENTATION

Atakan Ayyıldız
21526681

Ege Çınar
21627136

Ahmet Kasım Toptaş
21627667

ABSTRACT

Data scarcity can be the biggest limiting factor for achieving a good classification accuracy in machine learning problems. Here, we are trying to overcome this limitation by creating synthetic data using Generative Adversarial Networks (GANs). We train GANs with COVID-19 Chest CT Scan Dataset to investigate the effect of using GAN-generated images as a data augmentation technique. We use Convolutional Neural Networks (CNNs) to predict whether the patient has been infected by COVID-19. We got 80% test accuracy for the ResNet and VGG models with transfer learning with our augmented data which created by our GAN.

1 INTRODUCTION

It is significant to have enough data instances to achieve good results in machine learning field. Also the data should be evenly split among classes to prevent over representation of the classes with larger amounts of data. In some domains, it is harder to acquire adequate number of data instances due to either privacy concerns or difficulty of collecting data. These problems can be solved by generating synthetic data with a generative model such as Generative Adversarial Networks.

GAN is the most successful generative model developed in recent years and has become one of the hottest research directions in the field of artificial intelligence. The usage of GANs include data simulation, data augmentation for small dataset, style transformation, and gene data simulation.[1]

2 RELATED WORK

Loey et. al. [2] used a deep transfer learning model with classical and data augmentation and Conditional Generative Adversarial Networks to augment the data on COVID-19 chest CT scans. They tried to counteract the scarcity of the data as their dataset consisted of 760 CT scans.

They used five different deep convolutional network architectures(AlexNet, VGGNet16, VGGNet19, GoogleNet and ResNet50) to investigate the performance of the data augmentation techniques. Their generator network consisted of 4 transposed convonutional layers and the discriminator network consisted of 4 convolutional layers. They achieved good classification accuracy despite the limited size of their dataset.

Henrique et al. [3] gives details about two main criteria as to why we need to generate synthetic dataset, first one is the privacy concerns and the second is under-representation occurs (minor sampling) in class instances. According to the paper GAN is a good way to prevent over-fitting and create new content based on the training set. Oversampling decreases the accuracy but gives better precision. This means that oversampling on unbalanced dataset such as fraud detection gives better results.

In this study by Frid-Adar [4] et al in 2018, liver lesion classification was made using deep learning. There were cysts, metastases and hemangiomas as categories. Since the number of data is low, around 180 as medical images, the number of samples was increased by using classical data augmentation and liver lesion classification was made using CNN with this data. As classical data augmentation, they applied translation, scaling, flipping and shearing, which includes affine transformations, to these medical images in gray-scale images [5], [6].

Afterwards, synthetic data was created by using GAN methods with the data obtained from classical data augmentation and a comparison was made with the previous results using CNN. They obtained higher accuracy, sensitivity and specificity in the classification using GAN. Sensitivity increased from 78.6 percent in using only classic data augmentation, to 85.7 percent in best GAN method. ACGAN and DCGAN were used as GAN methods.

They obtained better results in terms of both sensitivity and specificity for DCGAN. In addition, they had the quality of their augmented data checked by radiologists. They also tried these methods (using GAN and CNN) in the state-of-the-art BoVW-MI[7] method and their methods gave better results than the BoVW-MI method. In conclusion, they believe this approach (classical data augmentation + GAN + CNN) helps other medical classification practices and supports radiologists' efforts to improve diagnosis.

3 METHOD

3.1 Generative Adversarial Network

Generative Adversarial Networks (GANs) were proposed by Ian Goodfellow in 2014. GANs consist of two artificial neural networks working against each other namely discriminator and generator. Discriminator network is trained to distinguish real images and fake images generated by the generator network. Generator network is trained to map points in the latent space to create new instances of the data.

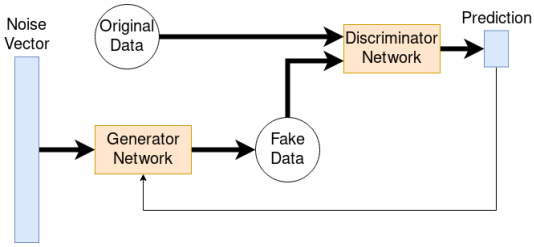


Figure 1: How a GAN trains both generator and discriminator network at the same time

The generator network is evaluated by the discriminator network and the loss is calculated as the difference between the prediction of the discriminator and 1. This means the loss will be low when the discriminator network predicts the generated image as a real and vice versa. The discriminator loss is calculated as the difference between predicted labels and real labels. The label indicates whether the image is generated or real.

The generator network in our GAN model consists of a fully connected layer with 262144 nodes followed by Leaky ReLU activation function with negative slope of 0.2. 2 transposed convolutional layers with the filter size 4 and stride 2. 2 Leaky ReLU activation layers with the same slope and a convolutional layer with filter size 7, stride 1 and Sigmoid activation function. There are 128 filters in each of transposed convolutional layers. [3]

The discriminator network consists of 2 convolutional layers with filter size 3 and stride 2. 2 LeakyReLU layers with the negative slope of 0.2 and 2 dropout layers with the probability of 0.4. At the end there is a fully connected layer with one node and sigmoid activation function. All convolutional layers have 64 filters. Adam optimizer with learning rate 0.0002 and binary-cross entropy as the loss function are used in both discriminator and generator networks.

3.2 Deep Transfer Learning (DTL)

DTL is a machine learning technique, where information obtained while training was held in one type of problem will be used to train in similar type of problem. Generally it is hard to have sufficient dataset to train whole CNN from scratch. Therefore DTL is commonly used as pre-trained models in deep learning methods such as computer vision and NLP. After pre-train on deep learning network, network is ready to evaluate. DTL can also be used as feature extractor and works well in different dataset scenarios using fine-tuning. (When dataset is small/large and similar/different)

3.3 Conditional Generative Adversarial Networks (CGAN)

CGAN first introduced by Mirza et al[9]. CGAN uses a supervised learning approach takes the random noise "z" and the class label "c" as inputs to the generator, then generates the fake outputs

with its label. Then the discriminator takes these outputs as the inputs and makes a correlation between labels and images. Network introduced another conditional variable "y" into both generator and discriminator as an extra input. According to the paper[9] by adding additional conditional class label it is directly possible to adjust the data generation process.

3.4 Convolutional Neural Network

CNN is widely used in computer vision and is very important for image classification. We want to classify our test data using CNN, the synthetic pictures that we created with GAN and the original data we had before. In the articles we read, smaller models and less complex models are generally used for medical imaging due to smaller image size. But we are considering using a slightly more complex model due to our larger image size. While doing our classification, we used gray images, so we used single channel input. We do this to train our model easier and faster. Figure (2) represents our CNN model.

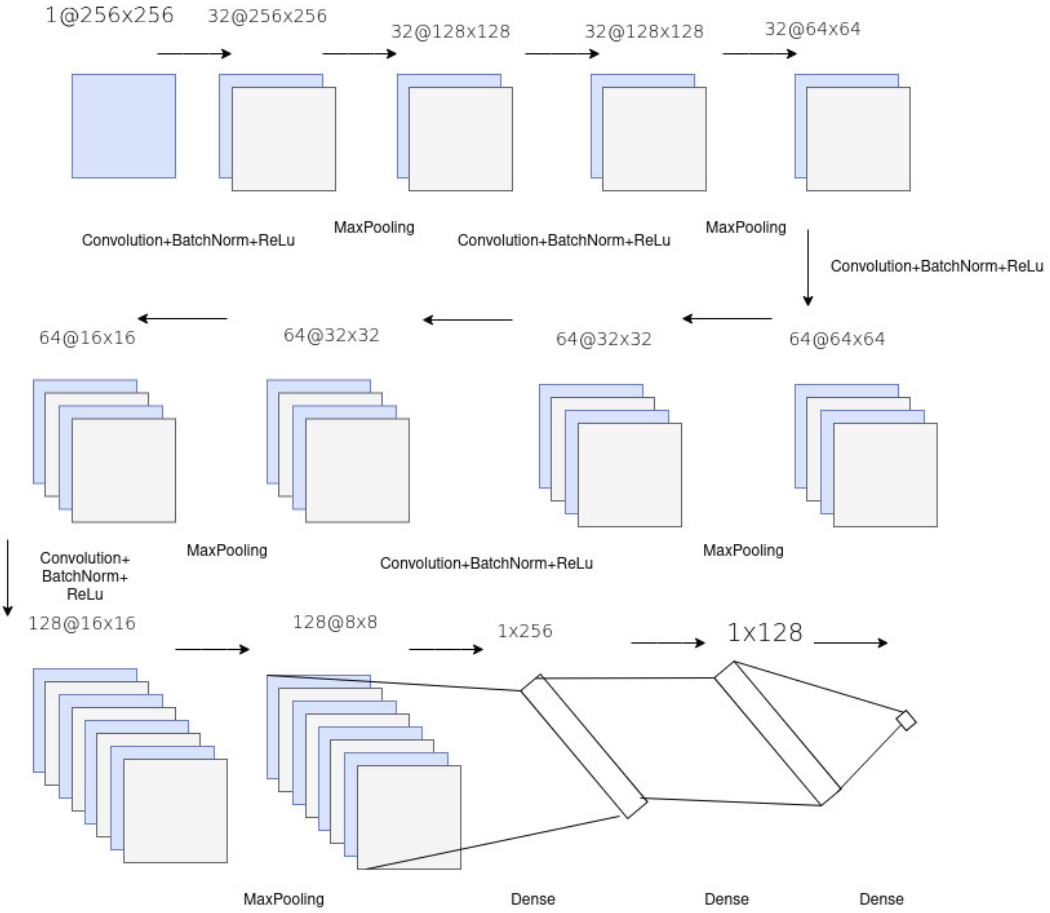


Figure 2: CNN Architecture

We have 5 blocks in our model and our blocks have a Convolutional layer and a MaxPool layer with activation function. Our ordering is Convolutional layer, activation function and Max Pooling. Our activation function is Rectified Linear Unit(ReLU). Our Convolutional kernel size is 3x3 and we use 1 stride and 1 padding for each Convolutional filter. We will have number of Convolutional filters as 32,32,64,64,128 in order.
Our kernel size is 2x2 and stride is 2 for the MaxPooling. Before each ReLU, we do batch normalization. Since we did not normalize the data at first, we wanted to use batch normalization. Ioffe and Szegedy [8] introduced batch normalization, and we used batch normalization before the activation function because they used batch normalization before the activation function.

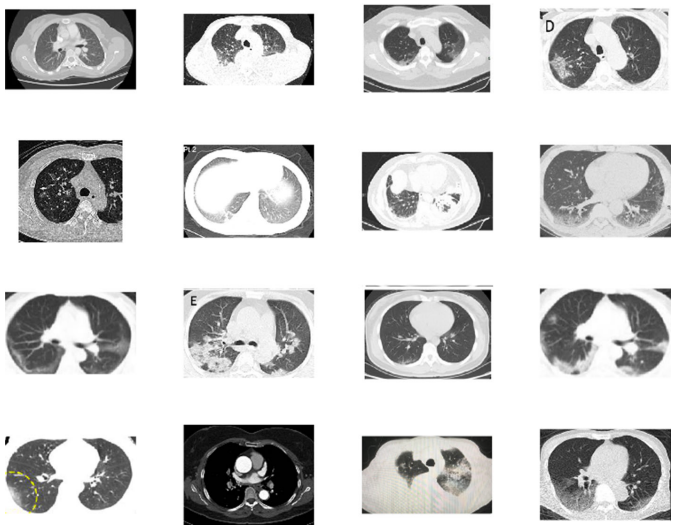


Figure 3: Samples of the used COVID/NonCOVID CT images

After 5 blocks, we have 2 dense layers. Our first layer consists of 256 neurons, 2. Our second layer consist of 128 neurons and our output layer consist of 2 neurons. To reduce the over-fitting, we added a dropout layer after the first dense layer and its probability is 25%. Our output activation function is Softmax. So we used ReLu as activation function except for the output layer.

While training our CNN model, we used batch size of 32 and a learning rate of 0.001. Our epoch size is not fixed, so we trained our CNN model until our train accuracy was above 95 percent and used the parameters with the best validation loss as the model. So we used checkpoint. We used SGD optimization as. Our loss is Categorical Cross-entropy.

4 EXPERIMENTAL SETTINGS

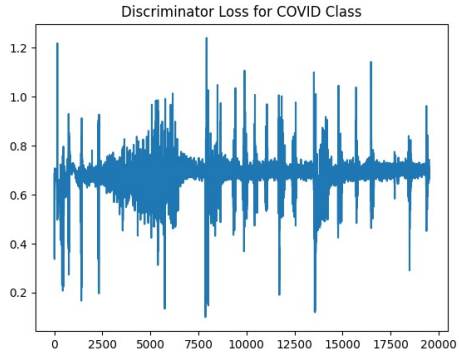
Chest CT images are good for the classification of COVID-19. The images are collected from different papers which are introduced in the related paper. Due to privacy rules only limited sized data is available to the public. Only 742 images are presented, 345 images are Covid, 397 images are Non-Covid which means the original dataset is small. So that's why we are introduced our problems. We are generating synthetic images with generator and authenticate with discriminator using GAN methods. After an image accepted it is added to the dataset for future evaluation.

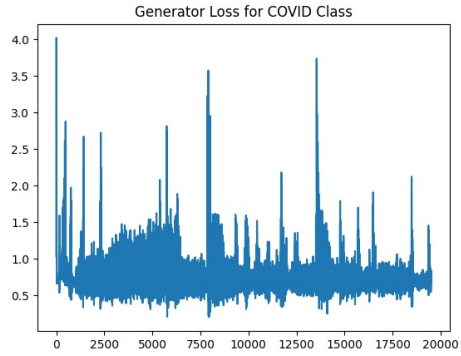
Model evaluation was handled by using tensorflow and keras libraries. It is simple and powerful to generate synthetic dataset. But it requires large computation power. Therefore we use Google Colab and Kaggle for GPU usage.

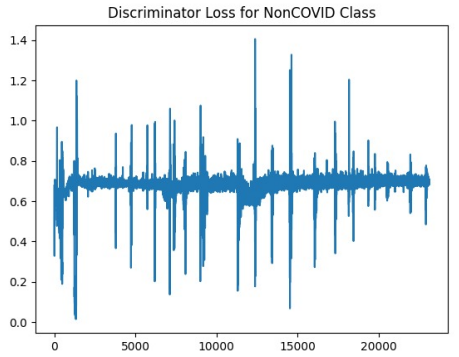
Firstly as we have already stated before our dataset is too small for that processing. So that we first create synthetic images for oversampling with GAN methods. In order to create synthetic images we use whole dataset with 3300 epochs. After creating the synthetic images, we split the original dataset into train,validation,test sets then we train our network and test it with both original dataset and synthetic dataset. This is the first phase.After that we do the same training process on the synthetic dataset with dividing it into train,validation,test synthetic sets. Then we compare the accuracy, recall and precision results.

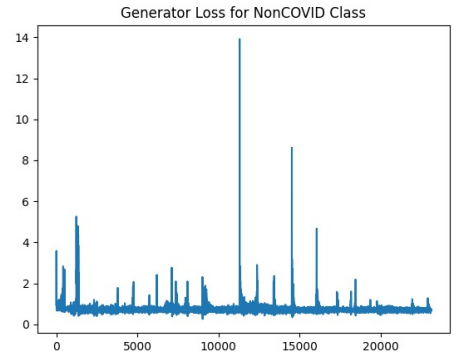
5 EXPERIMENTAL RESULTS

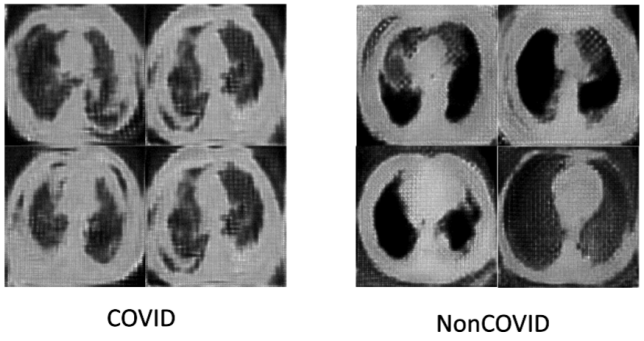
Both GANs are trained for 3300 epochs. We periodically save the generator models and images in order to compare and generate synthetic images with selected models. Models are saved every 300 epochs. Loss plots of discriminator and generator network are shown below. The outliers in loss might show the training process did not go smoothly. Some samples from generated images are shown below. COVID and NonCOVID images are distinctly different.











We trained the original data as a baseline using the proposed CNN model and we got almost 65.83 percent accuracy. To improve the accuracy we tried a deep transfer learning model with VGG-16 architecture. We added fully connected layers at the end of the network. We added a dropout layer with 25 percent chance in between the fully connected layers to decrease over-fit. We trained the original data with the VGG-16 and got 75.38 percent accuracy. We also want to use Resnet50 model. So we got 0.7638% accuracy for the test set. We then used the classical data augmentation with only horizontal and vertical flips and 45 degree rotations. We trained both networks with only classical

data augmentation and got 0.7186 accuracy for our CNN, 0.7839 accuracy for the VGG-16 and 0.7688 accuracy for the Resnet50.

To compare with the paper we read, we added the GAN generated images from LOEY[2] to the original dataset. There are 2000 synthetic images from both classes. We trained the our CNN and VGG-16 with these images and got 70.35 percent and 73.37 accuracy.

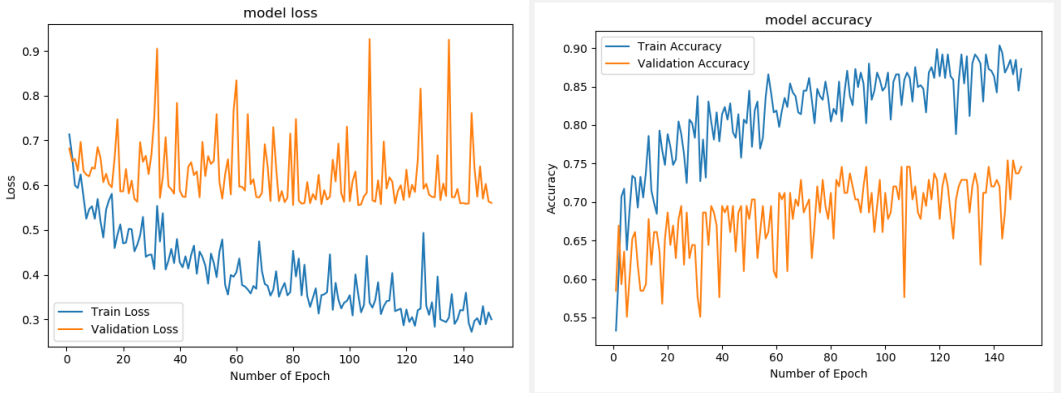
We finally combined these techniques and trained our models with both GAN generated images and classical data augmentation.

Model	Raw Data Test Acc.	Raw+Classic Aug. Test Acc.
Our CNN	0.6583	0.7186
VGG-16	0.7538	0.7839
Resnet50	0.7638	0.7688

Model	Raw + Our GAN Aug. Test Acc.	Raw + Loey Gan Aug Test Acc .
Our CNN	0.6985	0.7035
VGG-16	0.8090	*0.7038
Resnet50	0.8040	*0.7657

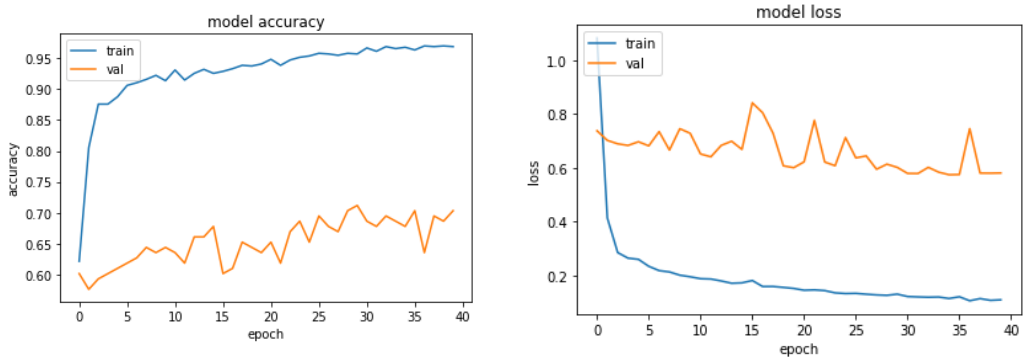
Our starred(*) values are taken directly from the paper we read[2].

Following figures shows model loss and accuracy of the best model which has the best test accuracy with our augmented GAN data. It is the VGG-16 model. We trained our VGG16 with 150 epochs but we got best parameters with best validation loss value. Its test accuracy is 0.8090% and test loss is 0.5202 with categorical cross-entropy.



We normalized our images for VGG16 tasks but since we have batch normalization layers in our CNN models and Resnet50 models we didn't normalize. We used 32 as batch size and SGD as the optimizer for all prediction tasks. We used learning rate as 0.001 for our CNN models and 0.005 for the deep learning transfer models which are Resnet50 and VGG16.

Following figures shows model loss and accuracy of the Resnet50 with our augmented GAN data. We trained our Resnet50 with 50 epochs but we got best parameters with best validation loss value. Its test accuracy is 0.8040% and test loss is 0.5476 with categorical cross-entropy.



6 DISCUSSION/CONCLUSION

We successfully increased classification accuracy by adding GAN generated data to our dataset. This technique can be used when the data is scarce or when the nature of the dataset compromises the privacy of individuals. This problem can be solved by only using generated data instead of the original data.

We struggled with creating quality GAN images. In generated images we had a hard time replicating the white areas in the original images. We selected 250 good quality images from generated images to add to our augmented dataset.

As definition of GAN, high resolution synthetic can be produced from low resolution images. GAN based models provides a good solution for lack of data. However still GAN have some limitations. For instance legs and head CT images can not be generated well yet. The main reason these limitations' occur is because the GAN is still in its early stages. Even though these limitations, GAN is going to be more extensive in biomedical researches. According to Frid-Adar et al.[4] GAN models have better accuracy than traditional data augmentation. GAN accuracy will be relatively low when the dataset is small. Since GAN uses generators of deep learning methods, not setting them up properly will often give erroneous outputs.

The latent space is simply a representation of compressed data in which similar data points are closer together in space and latent space is considered as a bottleneck. Latent space size can be changed to change quality of the generated images. But the speed of the training vary on the dimension of the latent space.

In order to avoid overconfidence, when training on the discriminator, label smoothing is a good technique. Max pooling should be avoided for down sampling and convolution with stride should be used. If time is not a constraint, then pooling layer should not be used and convolutional layer should be used to not reduce the variance in the generated images.

REFERENCES

[1] Lan, You Lei, Zhang Zeyang, Fan Zhiwei, Zhao Weiling, Zeng Nianyin, Chen Yidong, Zhou Xiaobo. *Generative Adversarial Networks and Its Applications in Biomedical Informatics, 2020.*.

[2] Loey, M., Manogaran, G. & Khalifa, N.E.M. *A deep transfer learning model with classical data augmentation and CGAN to detect COVID-19 from chest CT radiography digital images. Neural Comput & Applic (2020).*.

[3] Fabio Henrique Kiyoiti dos Santos Tanaka and Claus Aranha. *Data Augmentation Using GANs. In Proceedings of Machine Learning Research, 2019.*.

[4] Frid-Adar, M., Diamant, I., Klang, E., Amitai, M., Goldberger, J., Greenspan, H. *Gan-based synthetic medical image augmentation for in-creased cnn performance in liver lesion classification.arXiv preprint arXiv: 1803.01229, 2018.* .

[5] H. R. Roth, L. Lu, J. Liu, J. Yao, A. Seff, K. Cherry, L. Kim, and R. M.Summers. *Improving computer-aided detection using convolutional neural networks and random view aggregation,"IEEE Transactions onMedical Imaging, vol. 35, no. 5, pp. 1170–1181, May 2016.* .

[6] Setio, A. A. A., Ciompi, F., Litjens, G., Gerke, P., Jacobs, C., van Riel,S., Wille, M. W., Naqibullah, M., Sanchez, C., van Ginneken, B. *Pulmonary nodule detection in CT images: false positive re-duction using multi-view convolutional networks. IEEE Trans MedImaging 35 (5), 1160–1169., 2016.* .

[7] I. Diamant, E. Klang, M. Amitai, E. Konen, J. Goldberger, and H. Greenspan. *Task-driven dictionary learning based on mutual in-formation for medical image classification," IEEE Transactions on Biomedical Engineering, vol. 64, no. 6, pp. 1380–1392, June 2017.* .

[8] S. Ioffe and C. Szegedy. *Batch normalization: Accelerating deep network training by reducing internal covariate shift.InProceedings of ICML,jmlr.org/proceedings/papers/v37/ioffe15.pdf, pages 448–456, 2015.*.

[9] M. Mirza and S. Osindero. *Conditional generative adversarial nets. arXiv[Preprint]. (2014):2672–80. arXiv:1411.1784*.