On Data Augmentation Using GANS

Divya Guruswamy, Rhea Sukthanker, Peter Gronquist

Abstract—**In recent times Generative Adversarial Networks (GANs) have gained huge popularity, which is mainly attributed to their more generic ability to generate new high resolution images. GANs have many applications like photo blending, in-painting, photo-editing etc. Additionally it has also been claimed that GANs could be effectively used to augment a dataset to improve the accuracy of a classifier. This is particularly useful when data and label collection is expensive. In this project we use class conditional GANS like (AC-GAN, CGAN) to augment the dataset with new images. In addition we perform some modifications on the AC-GAN objective, using pretrained z vectors from ALI (Adversarially Learned Inference) GAN. In general we observe that the classification performance gain obtained by using GANs depends on the dataset type and the amount of data per class available to the Generator. We obtain up to 1% of performance gain on the medical dataset.**

I. Introduction

It is often claimed, that nothing beats just applying more data in deep learning. More data (from a particular class) becomes even more important when the available dataset suffers from a severe class imbalance. Unfortunately, the issue of imbalance is very common, especially in the medical domain, where collecting more data is extremely expensive and difficult. In such cases, the use of data augmentation to artificially increase the size of the available data, is a very attractive solution. This additional augmented data is then mainly based on effective modifications to the available training data. The most common augmentation method used in traditional computer vision literature so far was the affine transformation of natural images. The recent deep learning era has widened our options for data augmentation [1]. Using deep generative models like GANs [2], we can now generate new data that is far away from only an affine transformation of existing data. There have, however, been quite contrary claims made as to whether using a GAN for augmentation actually helps the classifier. In this project, our aim was to study the effects of data augmentation using different GAN variants, with novel modifications, on the classification accuracy for a simple neural network classifier.

II. Related Work

Given that procuring the right data in large enough quantities is considered one of the hardest tasks in data science, especially in the medical field, there have been many works on finding ways to augment data that already exists. In fact, regular (not using neural networks, e.g. random cropping, rotation, scaling, noise injection etc.) data augmentation has already become the norm for many works. But with the emergence of new and different kinds of GANs a whole new field of data augmentation has opened itself up to researchers. A recent work doing similar augmentation to ours but applied on liver lesion classification [1] has already successfully achieved better classification accuracies using GAN-augmented samples. Similarly, GANs and CycleGANs have also been applied to CT segmentation tasks [3] sample augmentation and have shown stark improvement, compared to non augmented data. Based on all of these successful works, we explore the efficiency of similar and newer GAN sample augmentation methods applied to a new breast histopathology dataset [4], as well as different standard datasets in the following sections.

III. Datasets

For Data Augmentation we consider the following three datasets, Fashion MNIST[5], CIFAR100[6] and a breast histopathology dataset[4]. The choice of these datasets is mainly motivated by their different characteristics. Our first two datasets do not suffer from class imbalance, so we introduce one artificially, by sparsifying a few of their respective classes. The last dataset is naturally sparse and representative of a crucial real-world application.

1) Fashion-MNIST: Fashion-MNIST is a dataset of 28x28 grayscale images consisting of pieces of clothing, divided into 10 classes. It was intended to serve as a direct drop-in replacement of the original MNIST dataset for benchmarking machine learning algorithms. It includes a training set of 60,000 samples and a test set of 10,000 samples[5]. It provides us with a simple, low resolution opportunity, to test out the generative powers of our GANs.

2) CIFAR100: CIFAR100 is a collection of natural images divided into 100 classes. It consists of 60,000 32x32 colour images, with 600 images per class. There are 50,000 training images and 10,000 test images, making it 500 training images and 100 testing images per class. The 100 classes in the CIFAR-100 are grouped into 20 superclasses. Each image comes with a "fine" label (the class to which it belongs) and a "coarse" label (the superclass to which it belongs)[6]. With this dataset, we can test the GANs on more intricate images, with a smaller amount of data per class. Therefore making it a significantly harder problem.

3) Breast Histopathology: The original dataset consisted of 162 whole mount slide images of Breast Cancer (BCa) specimens scanned at 40x. From that, 277524 patches of size 50 x 50 (Colored, see Fig. 1) were extracted (198,738 invasive ductal carcinoma (IDC) negative and 78,786 IDC positive)[4]. Our task at hand is to classify the specimens as positives (class 1) or negatives (class 0). Given the size of the dataset we see that there is a clear class imbalance problem, making it an optimal real world case to test our GAN augmentation on.

(a) Cancerous cell

(b) Non Cancerous cell

Fig. 1: Breast Histopathology Dataset examples

IV. Baselines

A. Traditional Data Augmentation

Our first baseline uses traditional affine transformations of a given image to generate the required augmented dataset. More precisely, the transformations we use are: Rotation, rescaling, horizontal/vertical translation and horizontal flip (See Fig.2). This is often a common practice in computer vision literature and is known to improve the accuracy of deep classifiers [7]. The motivation behind this baseline is that an image should be classified correctly regardless of the affine transformation applied to it beforehand. Thus, using affine transformations, we can artificially increase the size of the already available dataset.

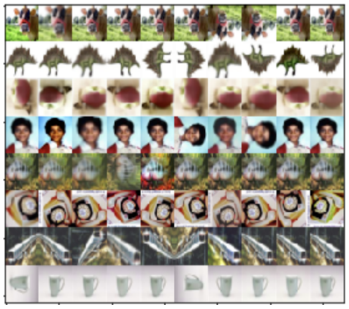


Fig. 2: Examples of augmented images from the CIFAR dataset

B. GANs for Data Augmentation

Generative Adversarial Networks(GANs) were introduced with the aim of generating new data using some already available data. A GAN is usually composed of a Discriminator and a Generator. The objective of the Generator is to generate data so as to fool the Discriminator into believing that the data is real, while the objective of the Discriminator is to identify the generated data from the real data. Thus intuitively a GAN is a two player adverserial game between the Discriminator and the Generator. Therefore, the ideal goal is to end up at a saddle point between the two players. Mathematically the objective of GAN in the most general case is as follows :

$$\min_{G} \max_{D} V(D, G) \tag{1}$$

where,

$$V(D,G) = E_{\mathbf{x}\sim p_{\mathbf{X}}}[\log D(\mathbf{x})] + \underbrace{E_{\mathbf{x}\sim p_G}[\log(1 - D(\mathbf{x}))]}_{E_{\mathbf{z}\sim p_{\mathbf{Z}}}[\log(1-D(G(\mathbf{z})))]}$$

Given that a pretrained Generator is able to generate random data, it is a great candidate for usage in data augmentation. More effectively even, there are class conditional variants of GANS (eg: CGAN, ACGAN) which can specifically be trained to generate images belonging to a particular class. Next we briefly describe the types of GANs we considered for our data augmentation task.

1) CGAN: In the most general case, GAN models are able to generate new random possible examples for a given dataset. They do not allow us any control over the type or class of images we generate. This is a roadblock for data augmentation, as to augment an existing dataset with some training images we need an (image, label) pair which cannot be obtained from a traditional GAN model. One way to deal with this issue, is by using a CGAN [8] (Conditional GAN). In this model the image generation can be conditioned on a given class label. The objective of the CGAN is same as that of a traditional GAN. The only difference is that now we additionally have a dense embedding vector corresponding to every class, which serves as a sort of stimulus for the model to generate samples, conditioned on a particular class.

2) AC-GAN: Another solution to overcome the inability of a traditional GAN to generate images of a specific class, is to use the AC-GAN [9] (Auxiliary Classifier - GAN). An AC-GAN can be seen as a modification to the CGAN mentioned previously. An issue with the CGAN is that, though we have a notion of class now, we still do not ensure explicitly that the images generated by the generator are classified correctly to the same class. This is mainly because the loss function of the model still only includes the real-fake loss as in a traditional GAN. Keeping the loss function the same is based on the assumption that the task of real-fake detection and classification are quite similar to each other. This is often not the case, hence an AC-GAN adds an additional cross-entropy loss term to the objective function of the CGAN. Thus we are now guaranteed to generate images belonging to a class, which are realistic and also classified correctly. The Generator accepts a noise sample and a label to generate a fake image, $X_{fake} = G(z, c)$. The discriminator accepts an image and produces two outputs $D(x) = P(S|X), P(C|X)$. The real-fake log likelihood (L_S) and the cross-entropy log likelihood for classification (L_C) are then calculated as :

$$\begin{aligned} L_S = E[\log P(S = real|X_{real})] + \\ E[\log P(S = fake|X_{fake})] \end{aligned} \tag{2}$$

$$\begin{aligned} L_C = E[\log P(C = c|X_{real})] + \\ E[\log P(C = c|X_{fake})] \end{aligned} \tag{3}$$

The Generator's objective is $\max(L_C - L_S)$ while the discriminator's is to $\max(L_S + L_C)$. The architecture of AC-GAN is represented in Figure 3.

V. Proposed Modification

A. AC-GAN proportion sampling

The classification loss of AC-GAN drives the generator to produce images which have a low classification loss. However this loss is calculated by the discriminator and depends only on the generated images. It does not give any direct indication of the classifier's loss. Ideally, one would like to train the

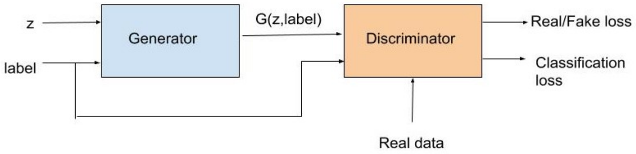


Fig. 3: AC-GAN architecture

generator to optimize the classifier's loss on the augmented data directly, to guarantee an increase in accuracy. However, the generator can be optimised only by the loss on the fake images and therefore cannot be trained on the augmented dataset.

The discriminator is trained with a batch of data containing 50% real data and 50% generated images. A possible modification is to train the discriminator with a batch which reflects an augmented dataset, so that the training matches how the classifier is trained. Hence when the generator minimizes the discriminator loss, it would be working towards reducing a loss of the discriminator, which is similar to improving the classifier's accuracy. Often the data augmentation is performed to balance a dataset. In this case, the percentage of real and generated images in the final augmented dataset would not be equal. The classifier would then be trained on this dataset which contains a larger number of real data. Thus, in the modified training of discriminator, the percentage of real data in the batch is higher than the percentage of generated data. We study the performance of two training modifications.

- AC-GAN_v2: Each batch of training data for the discriminator consists of 75% real images and 25% generated images.
- AC-GAN_v3: Discriminator is trained with 75% real images for most of the epochs. Every third epoch the batch contains 50% real images so that the discriminator occasionally uses more fake images to train.

Both the modified versions are stable and generate images similar to an AC-GAN with normal training. Another training modification which we explore is training with 25% real data and 75% fake data, which we call AC-GAN_v1. This model therefore sees less real samples. Training this model allows for the analysis of the variation in output with different training batch proportions.

The images generated for each dataset after the hybrid training(AC-GAN_v3) are represented in Figure 4. It is seen that images have variety and look similar to real data. The images for AC-GAN_v2 are present in the appendix.

B. *Modifying the AC-GAN with ALI*

Though the AC-GAN fulfills our purpose of obtaining augmented (image,label) pairs from the generator, we still have an issue. Note that our goal is not to improve the classification accuracy on the generated images, but to improve the accuracy on the full augmented dataset. Unfortunately there is no straightforward way to directly encode this desired behavior into our model. Initially we thought of encoding this behavior through mixing of the real and fake samples while training the Generator, but this does not work as there is no path for gradient propagation from the real samples to the generator (thus mixing samples has no effect on the training). Therefore the goal should be to somehow encode the fact that the images are real or fake implicitly through the input z to the Generator.

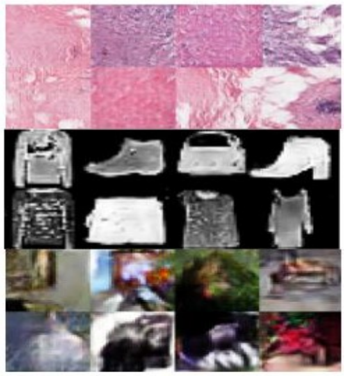


Fig. 4: AC-GAN_v3 generated images

Adversarially Learned Inference (ALI) [10] is a network composed of three players: the Discriminator ($D(x,z)$), the Generator ($G_x(z)$) and the Encoder ($G_z(x)$). The Generator samples z from a random normal and generates an image. The Encoder takes as an input an image, predicts a mean and variance for the input and maps it to a z (drawn from the predicted normal distribution). The task of the discriminator in this model is to distinguish between the joint space of (image,z). Here the real samples correspond to the (image,z) pair corresponding to the Encoder, and the fake sample corresponds to the (image,z) pair corresponding to the Generator. Together the Encoder and the Generator work towards fooling the Discriminator (See Fig. 5).

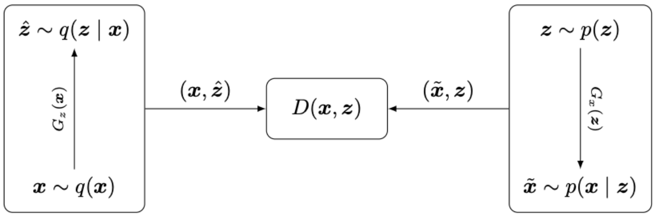

$\hat{\boldsymbol{z}} \sim q(\boldsymbol{z} \mid \boldsymbol{x})$
$G_z(\boldsymbol{x})$
$\boldsymbol{x} \sim q(\boldsymbol{x})$
$(\boldsymbol{x}, \hat{\boldsymbol{z}})$
$D(\boldsymbol{x}, \boldsymbol{z})$
$(\tilde{\boldsymbol{x}}, \boldsymbol{z})$
$\boldsymbol{z} \sim p(\boldsymbol{z})$
$G_x(\boldsymbol{z})$
$\tilde{\boldsymbol{x}} \sim p(\boldsymbol{x} \mid \boldsymbol{z})$

Fig. 5: ALI Architecture[10]

Once the ALI network is sufficiently trained, we can use the pre-trained Encoder to obtain the latent vector z corresponding to the real samples. ALI optimizes the following objective:

$$\min_{G_z, G_x} \max_{D} V(D, G_x, G_z) \tag{4}$$

where

$$V(D, G_x, G_z) = E_{\mathbf{x} \sim p_{\mathbf{X}}} \Big[\underbrace{E_{\mathbf{z} \sim p_{Gz}(\cdot|\mathbf{x})} \left[\log D(\mathbf{x}, \mathbf{z})\right]}_{\log D(\mathbf{x}, G_z(\mathbf{x}))} \Big] +$$
$$E_{\mathbf{z} \sim p_{\mathbf{Z}}} \Big[\underbrace{E_{\mathbf{x} \sim p_{Gx}(\cdot|\mathbf{z})} \left[\log\left(1 - D(\mathbf{x}, \mathbf{z})\right)\right]}_{\log(1 - D(G_x(\mathbf{z}), \mathbf{z}))} \Big]$$

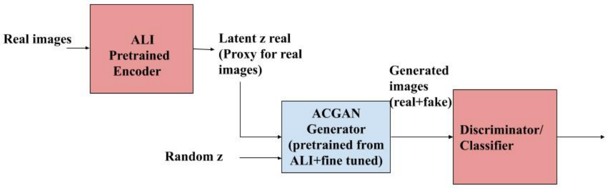

Real images
ALI Pretrained Encoder
Latent z real (Proxy for real images)
Generated images (real+fake)
ACGAN Generator (pretrained from ALI+fine tuned)
Discriminator/ Classifier
Random z

Fig. 6: AC-GAN modified Architecture

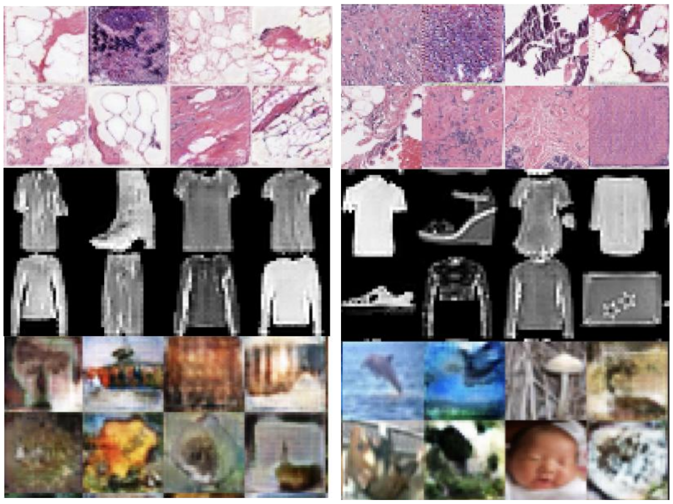
Additionally, using these vectors z (as a proxy for real samples obtained from ALI) combined with random z (for fake samples) to train the Generator, we now are able to encode the fact that we aim to improve the accuracy on the full augmented dataset (including the training set). Note that for the z corresponding to the real samples (obtained from ALI) we need to mask the real-fake loss, to ensure that real samples do not contribute to this loss. In the section below we explain the architecture in greater details.

1) AC-GAN based on ALI: In our first model, the input to the Generator of the GAN (while training the Generator) is a mixture of the real and fake latent vectors z. The real z are the ones obtained as output of the Encoder of the pretrained ALI network, and the fake z are just random normal vectors. The intuition behind doing this is that in some cases we provide input to the Generator, which corresponds to a compressed version (z) of the actual images. We also ensure that the architecture of the Generator of the AC-GAN is the same as the architecture of the Generator of the ALI network. This gives us two options while training the AC-GAN. The first one is to train the Generator from scratch while the second one is load the pretrained Generator from the ALI architecture and fine-tune it further. We discuss results obtained using both of these methods. Another choice we have is with regards to the training objective. Since the AC-GAN objective is composed of two components, we need to mask out the real-fake loss appropriately for the real samples, to ensure that they do not contribute to the real-fake loss of the model. Another possible choice is to avoid masking of the loss function completely. We discuss results using both these objectives in the discussion section. The architecture of the AC-GAN based ALI is as follows. The new masked real fake objective for the AC-GAN looks as follows:

$$L_C - L_S * 1(S = fake) \tag{5}$$

C. Examples of generated Images

Firstly, we can see that the reconstruction quality of the images is quite good to the naked eye (See Fig7). We attempted at using these reconstructed images to directly augment the data, but unfortunately, since the reconstruction is almost perfect, using these images has the same effect as replicating the data. Thus we obtain approximately the same classification accuracy. The first AC-GAN based on ALI model, which accepts as input a mixture of noise and real latent vectors z, generates images depicted in Figure 8. It is seen that the images are not as precise as the reconstructed images of ALI, however have quite a high clarity.



(a) ALI generated images. (b) ALI reconstructed images.

Fig. 7: ALI applied to our three datasets.

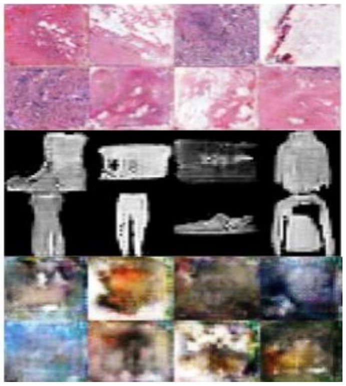


Fig. 8: ACGAN_ALI_v1 generated images

D. Classifier

We experiment with very simple Conv net based classifiers for the augmented and the unaugmented datasets [11]. These simple classifiers (tailored to image dimensions) are a combination of three convolutional layers followed by MaxPooling (with some dropout for regularization). The final layer is a Dense layer outputing the logits per class. The classifier is then trained with early stopping on validation loss, using the model with the lowest validation loss as final model. To train we use the Adam optimizer [12] implemented in Tensorflow with a learning rate of 0.001, a batch size of 64, on a categorical cross entropy loss.

E. Evaluation Metrics

We choose two metrics for evaluation of the classification accuracy: The regular accuracy and the top-k accuracy. For the breast histopathology classification we only use the regular accuracy, as it is a binary classification problem.

1) Accuracy: This is the simplest metric to evaluate the performance of the classifier. The metric is defined by the equation below

$$\text{accuracy} = \frac{\text{No. of correct predictions}}{\text{Total test cases}}$$

F. Top-k accuracy

$$\text{top-k accuracy} = \frac{\text{No. times correct class is within top-k}}{\text{Total test cases}}$$

VI. Results

The fashion MNIST and CIFAR100 datasets are first sparsified to create an imbalance [1] . The data is then split into training and test set. The GAN models are trained only on the train set. Finally, for each experiment, images are generated using the models for the sparse classes and augmented to match the number of real images, such as to balance the data set. Once the classifier is trained on the augmented training set, accuracy is calculated on the test set. Below are the results of using different models [2] for augmentation on the presented datasets.

Model	Fashion Mnist		CIFAR100		Medical
	Acc	Top-2	Acc	Top-5	Acc
No Aug	0.9157	0.98	0.5014	0.7756	0.8702
Affine	0.9112	0.9785	**0.5277**	**0.8116**	0.8756
CGAN	0.9173	0.9801	0.503033	0.7793	0.88
ACGAN	0.9178	0.9803	0.4662	0.7439	**0.8812**
ACGAN_v1	0.9152	0.9797	0.4809	0.7531	0.8803
ACGAN_v2	0.915	0.9799	0.4812	0.75583	0.8792
ACGAN_v3	**0.9201**	**0.9812**	0.48133	0.76	**0.88**
ACGAN_ALI_v1	0.9173	0.9806	0.5083	0.7815	0.8754
ACGAN_ALI_v2	0.9169	0.9809	0.4952	0.7715	0.8755

TABLE I: Comparison of GANs with baselines

VII. Discussion

Table I compares the accuracy and top-k accuracy of the classifiers for the different datasets. This test set accuracy is averaged over three runs of the models (to reduce variance effects). From the table it is observed that each dataset shows a different trend in the classifier accuracy. Given the diversity with respect to scale (Gray, RGB), dataset-sizes etc. this is expected.

For the fashion MNIST dataset, the accuracy of the imbalanced dataset is already quite high. The accuracy for different GAN augmentations closely resembles this value. Overall it seems that the ACGAN_hybrid model performs the best with a gain of about 0.5%. As pointed out by T.Pinetz et al.[13], the performance of GANs for data augmentation are usually only as good as the real data they were trained on. Since fashion mnist has a large number of samples for each class, the classifier could achieve high accuracy even with an imbalanced dataset. When it is presented with more data of low resolution, which is the generated data, the classifier does not learn anything new as the images do not contain any new features which might have helped the classifier improve. Possibly the classifier is now more robust to perturbations of input, but since no new knowledge about the features of the images is introduced the accuracy remains more or less the same. Although GANs do not improve accuracy for the fashion MNIST dataset, they do not harm the classifier's accuracy when present. The same can not be said for the CIFAR100 dataset though (where the GANs seem to have a detrimental effect on the accuracy).

Each class of the CIFAR100 dataset consists a maximum of 500 images and is thus very limited. When trained on such a limited datsets, GANs do not produce very realistic images. Understandably, the accuracy of the classifier then falls when augmented with our GAN generated images, as they are of poor quality and hinder the classification. However, affine transformations on the dataset helps make the network transformation invariant, and hence improves the network's classification accuracy for CIFAR100.

For medical data though, the accuracy is improved by GAN augmentation by approximately 1%. The generated images in the augmented dataset help balance the previously highly imbalanced dataset, and provide more samples for the classifier to train on in the positive class (the initial imbalance between classes is about 75% 25%). And even though these images have low resolution, they still improve the accuracy of the classifier.

The classification accuracy when the classifier is trained only on the generated images is 0.7937 for fashion MNIST and 0.7786 for medical images, which is much higher than a random prediction accuracy. Hence we can deduce, beyond the naked eye, that the images generated do in fact contain generalized information of classes.

VIII. Conclusion

Through our experiments we conclude that even though data augmentation using GANs seems promising, the performance gain obtained caps at about 1%. There could be several reasons for this. One major challenge with using GANs for data augmentation is deciding what kind of images are actually suitable for data augmentation (E.g. Resolution or underlying structure of represented data). Though it seems reasonable to choose the generator model producing the best looking images to the naked eye, it is still unclear whether choosing a generator producing slightly more noisier images would in fact help the classification accuracy more. Generally we can say that any form of image augmentation is actually useful, if the new images have certain properties/features which the classifier can potentially exploit. This raises an important question of whether GANs which basically learn from the already available data could potentially learn to benefit the accuracy of a given classifier, and learn of important features through this. There is clearly a need for more research addressing these questions, which we leave for future work.

References

[1] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, "Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification," *Neurocomputing*, vol. 321, pp. 321–331, 2018.

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[1] We sparsify 3 classes (ie. 1,3,7) in Fashion Mnist to 50%,65% and 80% of their original sizes respectively. For CIFAR we sparsify 20 randomly selected classes (sparsifing groups of four to 65%, 70%, 75% and 80% of their original sizes)

[2] All of our models were implemented in tensorflow version 1.13.1 and keras with tensorflow backend

[3] V. Sandfort, K. Yan, P. Pickhardt, and R. Summers, "Data augmentation using generative adversarial networks (cyclegan) to improve generalizability in ct segmentation tasks," *Scientific Reports*, vol. 9, 12 2019.

[4] "Breast histopathology dataset," https://www.kaggle.com/paultimothymooney/breast-histopathology-images.

[5] "Fashion mnist dataset," https://www.kaggle.com/zalando-research/fashionmnist.

[6] "Cifar-100 dataset," https://www.cs.toronto.edu/~kriz/cifar.html.

[7] A. Kwasigroch, A. Mikołajczyk, and M. Grochowski, "Deep convolutional neural networks as a decision support tool in medical problems–malignant melanoma case study," in *Polish Control Conference*. Springer, 2017, pp. 848–856.

[8] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[9] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 2642–2651.

[10] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville, "Adversarially learned inference," *arXiv preprint arXiv:1606.00704*, 2016.

[11] F. Sultana, A. Sufian, and P. Dutta, "Advancements in image classification using convolutional neural network," in *2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*. IEEE, 2018, pp. 122–129.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[13] T. Pinetz, J. Ruisz, and D. Soukup, "Actual impact of gan augmentation on cnn classification performance," 02 2019, pp. 15–23.

Appendix

The basic AC-GAN generates images as shown in Figure 9. From there, it is clearly visible, that the medical and fashion MNIST datasets have a higher clarity than the CIFAR100 images. This provides a visual explanation to the drop in accuracy for the CIFAR100 classification.

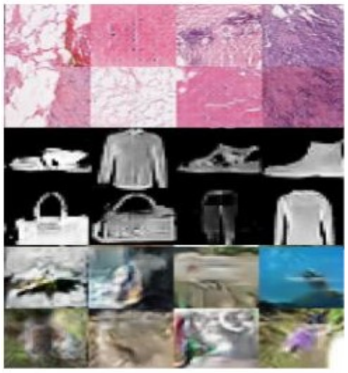


Fig. 9: AC-GAN generated images

AC-GAN_v2 which is trained with 75% real data and 25% fake data generates the images depicted in Figure 10. The images appear to have little more detailing than the AC-GAN images.

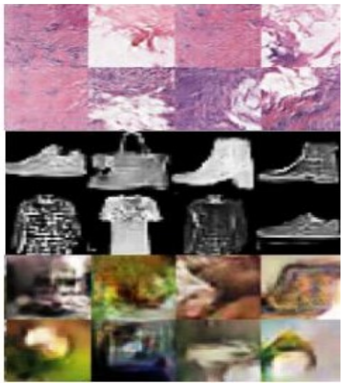


Fig. 10: AC-GAN_v2 generated images

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

On Data Augmentation Using GANs

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**
Guruswamy	Divya
Sukthanker	Rhea
Gronquist	Peter

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date

Zurich, 14/01/2020

Signature(s)

Divya Guruswamy

Sukthanker

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.