Data-Efficient GAN for Synthetic Samples of Rare Classes

Justin Oosterbaan, Robert-Jan Bruintjes, Attila Lengyel and Jan van Gemert

Abstract

Camera traps are used around the world to provide data on species, population sizes and how species are interacting. However this creates a lot of work in identifying which animal was actually spotted near the camera. Attempts have been made to use deep-learning to identify animals and work correctly for animals which are not rare but the lack of training data of rare species is a hurdle yet to be overcome. This research is focused how well the MIT Data-Efficient Generative Adversarial Network or MITGAN for short can generate realistic samples to be used as training data. For this we use a modified version of the CCT20 data-set. Which has artificially made a single rare class: the deer class. We trained a MITGAN model to generate images of our artificial rare class and used these images to train a classification model. This model was then compared to a baseline model as well as an oversampled model. From this it follows that using the MITGAN model to generate extra samples for our rare class is not worth the effort it takes compared to the precision increase in the rare class.

1 Introduction

Camera traps placed in the wild allow ecologists to perform a field-survey in a noninvasive manner, however this brings about a new problem. The large amount of images that have to be identified. To combat this problem as well as similar problems in different domains, substantial progress has been made in automatic visual recognition [1, 2, 3]. However, current visual recognition algorithms are unable to reach peak performance if there is a limited amount of training data [4]. One of the current solutions for this issue is to complement real data with ad-hoc simulated data [1]. In this research it was found that adding synthetic data can significantly decrease error-rates in classifying rare classes. However, only two similar methods of generating synthetic data were tested during the research, namely simulating a camera trap image by creating a virtual environment with a virtual deer in it.

In this research we wish to answer the question "How well can GAN generate realistic samples for rare classes?" [5]. We specifically use the MIT data-efficient GAN for this as it requires a significantly smaller training set compared to a regular GAN to gain similar results [5]. Considering the fact that one of the main issues we are trying to work around was that a low amount of training data existed. To answer this question there are a few sub-questions that I also wish to answer. One of which is how a deep-learning model trained with generated images of deer compares to a model trained with real images in identifying deer. Another sub-question is how well the same deep-learning model trained with generated images compares to a model trained with synthetic deer generated from the virtual environment mentioned earlier in this paragraph [1]. The questions are focused on comparing results when training a classification model as that is in line with the real life issue which this research was based upon.

2 Related Work

2.1 Generative Adversarial Networks

A Generative Adversarial Network (GAN) is a deep learning framework in which we train 2 models, a generative model that captures the data distribution and a discriminative model that estimates the probability that a sample came from the training data instead of from the generative model [7]. After the first GAN was proposed [7], researchers have found different methods of improving its performance and training stability such as more stable objective functions [5, 8, 9, 10, 11, 12], more advanced architectures [5, 13, 14, 15] and better training strategies [5, 16, 17, 18]. Thanks to these findings GANs have improved in both the visual fidelity aspect as well as being able to generate more diverse images. Most of these GANs require a large amount of training data to function properly, however one of these was specifically geared towards using small amounts of training data to gain comparable results [5], using a method they called "Differentiable Augmentation". With this method they were able to generate high fidelity images using as little as 100 images as training data. Considering the fact that we are trying to artificially increase our training data with GANs this seems to be the best solution to use.

2.2 Synthetic Samples

Instead of using GANs to generate data, researches have used simulation software to generate near-photorealistic images of animals with real-world nuisance factors such as a challenging pose, lighting and occlusions within the scene [1]. Using these synthetic images it was found out that the synthetic data can be used to considerably reduce error-rates for classes which are rare [1].

3 Methodology

To answer the questions posed for this research paper we have chosen a simple but rather time-consuming method. Following in the footsteps of the researchers at the California Institute of Technology [1], we selected the same data set to start our journey. The data set used is the Caltech Camera Traps (CCT) data set [6]. This data set contains 243,187 images from 140 camera locations with 30 distinct animal classes. More specifically we use the CCT-20 data split given in[6], a subset of CCT which contains 57,868 images from 20 camera locations with 15 distinct classes. In the CCT-20 data split the "deer" class has been artificially made into a rare class, only containing 44 images out of 13,553 total images in the training set. This data-set also defines **cis-locations** as locations seen in training and defined **trans-locations** as locations not seen in training. To focus on the performance of a single rare class, the other two rare classes from this data-set, namely the badgers and the foxes were removed [1] so to compare results on an even playing field, we did the same.

With the data set ready to be used, we start training the GAN using the images classified as "deer". More specifically we train the MIT data-efficient GAN [5]. This data-efficient GAN uses so-called "Differentiable Augmentation" [5] which allows the GAN to gain a substantial increase in accuracy while training with a small data-set compared to other GAN's [5]. Once the training is done we will have a model capable of generating images of deer.

Using this model we prepare our modified data-sets. Using the CCT-20 data-set with the other rare classes removed as a base we create 6 different extra data-sets for a total of 5. 2 of which are made by oversampling the deer in the data-set to add an extra 1000 and 10000 copies. The other 2 are made by using our generative model to generate an extra 1000 and 10000 samples for the deer class.

With these modified data-sets we train a classification model using the Inception-Resnet-V2 architecture. This model is first pre-trained on no-animal Imagenet which is a data-set defined as the Imagenet data-set with the "animal" subtree removed [1]. To keep the changed variables to only the data-set, we use an initial learning rate of 0.0045, RMSprop with a momentum of 0.9 [1, 19], and a square input resolution of 299. For data augmentation we use random cropping (containing at least 65% of the image), horizontal flipping, color distortion and blur. The model selection method is to use early stopping based on validation set performance. These are all the same hyperparameters as used in [1], which allows a fair comparison between results to be made.

4 Results

4.1 Data-efficient GAN

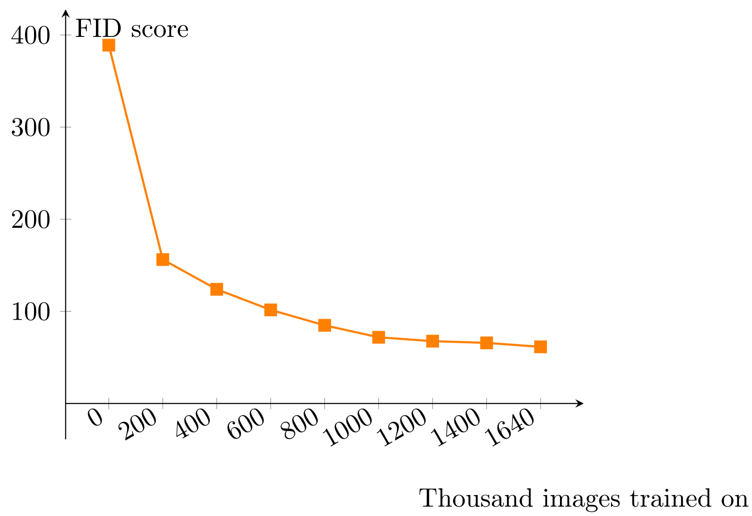
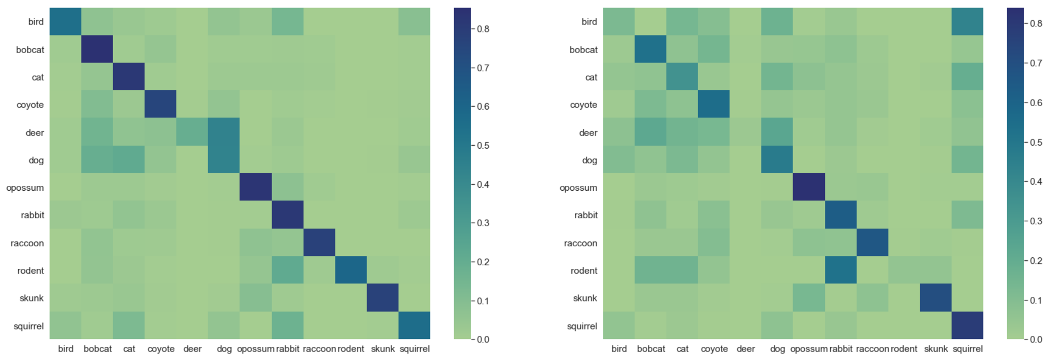


Figure 1: FID score of images outputted by GAN during training

We trained the Data-efficient GAN [5] with the small amount of deer images given in our data-set. After training the GAN for 5 days with a total of 1640 thousand images trained on the FID score was stabilizing and thus we stopped training here to avoid over-fitting.

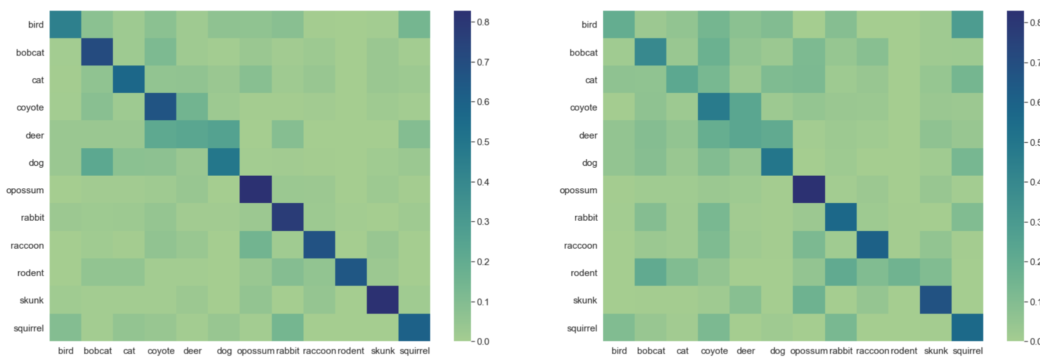
4.2 Classification

4.2.1 Confusion Matrices



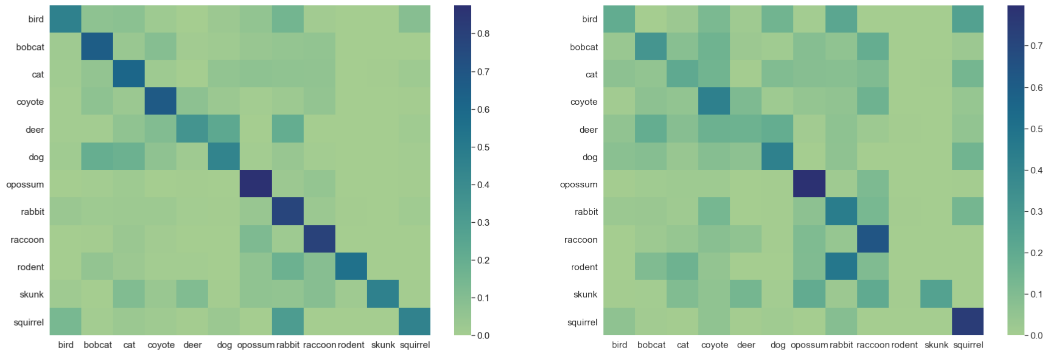
(a) Cis-locations (b) Trans-locations

Figure 2: Confusion matrix of model trained without extra images



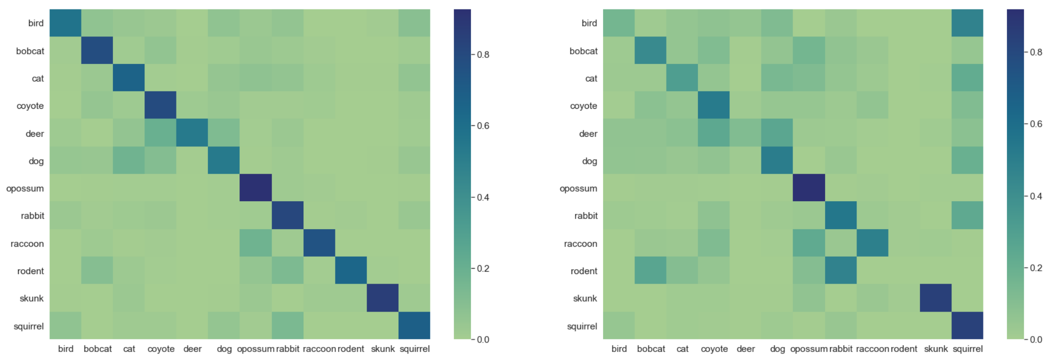
(a) Cis-locations (b) Trans-locations

Figure 3: Confusion matrix of model trained with 1000 generated extra images



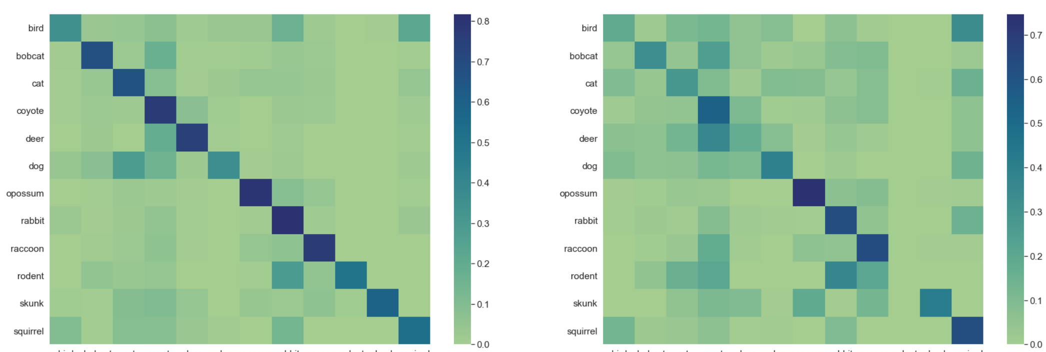
(a) Cis-locations (b) Trans-locations

Figure 4: Confusion matrix of model trained with 10000 generated extra images



(a) Cis-locations (b) Trans-locations

Figure 5: Confusion matrix of model trained with 1000 oversampled images



(a) Cis-locations (b) Trans-locations

Figure 6: Confusion matrix of model trained with 10000 oversampled images

Comparing all confusion matrices with each other we can see that in general the models do not have different classes which they often get confused with. Most of the confusion happens between the same classes. We can also see that the models are generally more confused when faced with locations never seen in training, which is to be expected.

4.2.2 Precision and Recall values

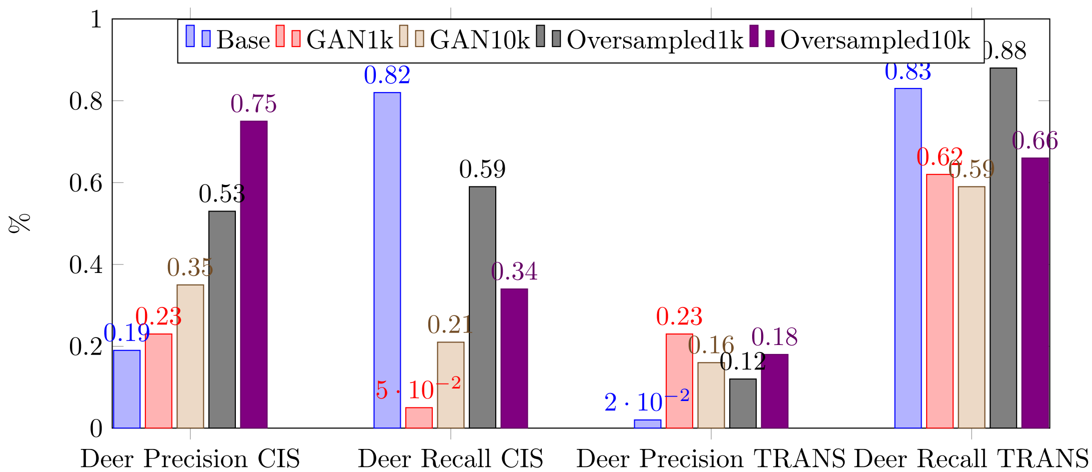


Figure 7: The Precision and Recall values of each model indicate that the over-sampled models outperform the GAN models

When comparing the precision and recall values of each model we notice a few things. Firstly in the locations which the model has seen before (CIS-locations) both the over-sampled models and GAN models perform better than the base model in precision but fall behind when it comes to recall. Secondly when faced with new locations (TRANS-location) the GAN1k model outperforms all other models on precision but performs second to last on recall. This means that a trade-off is required, one has to chose whether precision or recall is more important.

4.2.3 Average accuracy

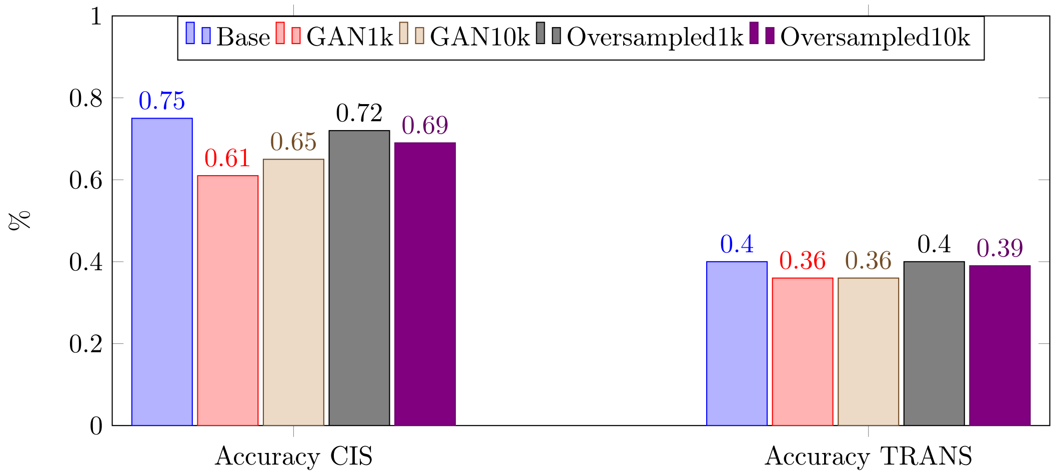


Figure 8: The average accuracy on both the CIS and TRANS locations indicate that both oversampling and using a GAN have a negative impact on the overall accuracy

For both the over-sampled models and the models with images generated by GAN added to the training data, the overall accuracy is lower than the accuracy of our baseline. Only on the TRANS-set is the total accuracy higher for the GAN10k model. This means that a decision has to be made over whether the added precision for the deer class is worth the lowered accuracy on all classes.

5 Responsible Research

5.1 Reproducibility

Reproducing a trained model perfectly is almost impossible for deep-learning, considering the fact that there is some randomness involved when training a model. However producing a result which is similar is possible if all variables are given. In this paper we have given these variables thus similar results should be able to be reproduced from this paper.

5.2 Ethical Responsibility

The ethical question for this research mainly is "Is using large amounts of electricity to train GANs for improving generalization for rare classes worth the added precision?" With the results we obtained we believe that it is not worthwhile to use this method for improving a classification problem. Considering the fact that the improvement is minimal and that there are other solutions which require less electricity and perform well. Such as using synthetic samples generated in Unity [1].

6 Discussion

Comparing our results to the results from [1] where they have increased the precision in the deer class on trans-locations by 39% whilst only reducing the accuracy of the other classes by 0.5% with our results where we increase the precision of the deer class by 21% and decrease the average accuracy by 4% we can easily see that for the trans-locations the synthetic samples used by [1] are superior in every way. For the cis-locations however their results do not outperform the baseline. In our case the GAN models do not outperform simple oversampling, thus we believe these results to be comparable with each other.

7 Conclusions

We have used the MIT-data-efficient GAN [5] in an attempt to improve the accuracy of classifiers specifically for rare classes. We trained a model with a data-set which contained images generated by said GAN and compared the results with a model trained by oversampling the rare class data as well as a model trained with the data-set without any modifications. Comparing these results we found that whilst the accuracy of the rare-class increases, the accuracy of the other classes have a significant drop. Comparing these results to the results from [1], the method of adding synthetic data generated by a simulation instead of a GAN. We see that the models trained on the simulation data significantly outperforms the models trained with data generated by a GAN. The use of a GAN to generate more samples also massively increases the training time of a classification model. Thus in conclusion, using the data-efficient GAN [5] to generate more samples is not worthwhile.

8 Future work

In this research we focused specifically on the MIT data-efficient GAN [5] for generating new images. This however is not the only GAN specifically for low-shot generation, maybe in the future this experiment can be repeated using a different low-shot GAN.

References

[1] S. Beery, Y. Liu, D. Morris, J. Piavis, A. Kapoor, M. Meister, N. Joshi, P. Perona. Synthetic Examples Improve Generalization for Rare Classes

[2] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun. Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639):115, 2017

[3] R. Poplin, A. V. Varadarajan, K. Blumer, Y. Liu, M. V. McConnell, G. S. Corrado, L. Peng, and D. R. Webster. Prediction of cardiovascular risk factors from retinal fundus photographs via deep learning. Nature Biomedical Engineering, page 1, 2018.

[4] G. Van Horn and P. Perona. The devil is in the tails: Fine-grained classification in the wild. September 2017

[5] S. Zhao, Z. Liu, J. Lin, J. Zhu and S. Han, Differentiable Augmentation for Data-Efficient GAN Training, 2020

[6] S. Beery, G. Van Horn, and P. Perona. Recognition in terra incognita. September 2018.

[7] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio. Generative Adversarial Nets, June 2014

[8] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein Generative Adversarial Networks. December 2017

[9] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. December 2017

[10] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. April 2017

[11] L. Mescheder, A. Geiger, and S. Nowozin. Which training methods for gans do actually converge? July 2018

[12] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. June 2016

[13] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. February 2018.

[14] T. Miyato and M. Koyama. cgans with projection discriminator. August 2018

[15] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. January 2016

[16] E. L. Denton, S. Chintala, A. Szlam, R. Fergus. Deep generative image models using a laplacian pyramid of adversarial networks. June 2015

[17] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. February 2018

[18] S. Liu, T. Wang, D. Bau, J. Zhu, and A. Torralba. Diverse image generation via self-conditioned gans. June 2020

[19] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural networks for machine learning, 4(2):26â31, 2012. 5

[20] Y. Skandarani, P. Jodoin, and A. Lalande. GANs for Medical Image Synthesis: An Empirical Study. May 2021.