FFTGAN: Generative Adversarial Network system for generating cosmological images in the frequency space

Alexander Lelidis, Khalid Aldughayem, Uros Tesic
group: TooOldTooComplex
Department of Computer Science, ETH Zurich, Switzerland

Abstract—**In this paper, we discuss techniques to quantify quality of cosmological images in the frequency domain. After selecting the neural network topology and input data for the discriminator that give us the best results, we discuss why the same data doesn't work for the generator. We also explain why the same approach fails for the generator and present our solution for that case. Finally, we discuss the optimizations used to generate large images.**

The problem consists of two parts. First, we had to find a way to quantify the concept of a "cosmological image". For this, we were given images and their scores. Second, we had to use knowledge and results which we obtained to generate a "cosmological image". The task proved to be hard in a few ways. A difference between a good and a bad image often is not detectable by the naked eye. Furthermore, the images are large in size, which required careful choice of techniques for discrimination and generation. Our solution achieves great grading performance. It is within 0.2 absolute error on the grading set. On top of it, we managed to generate relatively large images, which are similar to the prototype cosmological image. Our research show that classification (and grading tasks) on images which have strong local characteristics, which don't have a fixed global location (such as stars) can be accomplished better in the frequency domain. Furthermore, we obtain better results while using complex numbers in the neural network to represent FFT, than with two separate channels.

I. Introduction

Captured cosmological images by telescopes contain a lot of noise due to light pollution, cosmological background radiation, and multiple other factors. Taking a probabilistic viewpoint, a cosmological image can be described as a combination of the true star and noise distributions. Our objective is to learn the true star distribution and be able to compute a score which gives an indication to the quality of a given cosmological image. Furthermore, after learning the parameters for the true star distribution we want to sample that space and generate new cosmological images.

Our starting point is a data set of these images split up into two sections, scored images and labelled images. The scored image section contains a mapping from an image to a score $[0, 8]$, which indicates the quality of that image. The labelled image section contains labelled data into two categories (real cosmological image, fake cosmological image).

To address the first issue of computing a score for a cosmological image we relay on training a discriminator function in a supervised manner with the scored images. Recent publications [1] have shown that this objective is best addressed by a deep convolutional neural network. This network needs to be trained on a large collection of data to learn the complex mapping from noisy images to cosmological scores.

Considering that a construct of a "good cosmological image" is a highly irregular function we decided to use neural networks for the task. A Neural network consists of neurons, and dendrites (weights) connecting them. By changing these weights we can train the network to trigger different neurons for different inputs. With the increase in computing power in recent years, neuroscience, deep and convolutional neural networks have become increasingly popular.

The overall structure of the paper takes the form of seven sections, including this introduction. In Section II, we examine the relevant work to our contribution in this paper. Section III describes the models chosen, the methods used, and the motivation for following such an approach for each of the discriminator and generator. Section IV shows the results obtained, then they are discussed further in detail along with the limitations in Section V. Finally, Section VI gives a brief summary and critique of the findings, and areas for further research are identified.

Despite the fact that our method works on cosmological images, its functionality can be easily extended to other domains as well. Our main contributions can be summarized as follows:

- The first, to the best of our knowledge, discriminator that operates in frequency space and trained on real captured space images. Our method performs better on the regression task than most of the state-of-the-art methods.
- Incremental scaling of the generated image in the generator, and a probable explanation of why frequency domain generation of the image fails.
- Lastly, we analyse various parameters and design decisions in our framework. Including decoupling of complex numbers into channels and downsampling the input or tiling the images. To determine the optimal

hyperparameters we followed the gird search approach.

II. Related Work

There have been various approaches for generative models proposed in the last few years. Goodfellow et al. [2] proposed the first generative model, which is trained in an adversarial process. The basic idea is to train two models, a generator, which learns data distribution and a discriminator tries to distinguish between real and generated data. Radford and Metz [3] take that idea and combine it with the state-of-the-art deep convolutional networks and propose a strong candidate for unsupervised learning. By adding architectural constraints on the topology, the authors get a stable training under most settings. However, this models propose a more general viewpoint on the problem and are applicable to various domains. In the sense of cosmological data, Schawinski and colleagues [4] proposed a generative model trained on 4550 fits images from the Sloan Digital Sky Survey to recover features from degraded images.

III. Models and methods

A. Motivation

Before diving into the models, we took a closer look at the data set to motivate our design choices. We started by inspecting the labeled data set and realized that the "fake" and "real" images are actually very similar. In figure 1a we can see a real cosmological image and in 1b a fake one, which is actually quite different. However, figure 1c is also considered not cosmological image. We could not tell by just looking at images, which ones are "real" or "fake". This observation led us to the idea that we are not looking at the images in the correct basis. Inspired by that we decided to compute the Fourier transform of the images and inspect the representation in frequency space.

In figure 1e we transformed the good and the bad image into frequency space using the fast Fourier transform [5]. Here we can easily see a difference, while the good image has a wavey eye shape pattern, the bad one has most of its energy in the high frequencies, which represents the noise in the black regions and is not easily visible for the naked eye.

B. Method

For the scoring network, discriminator and the generator we use a two dimensional convolutional neural networks. We choose the convolutional 2D neural network because we are working on images and the convolutional neural networks are the best fit for that purpose[6]. The scoring network and the discriminator consist of several 2D convolutional layers (four in the case of the discriminator), each followed by a pooling layer to gradually decrease the resolution of the image. At the end we added a dense layer to aggregate and analyze global data from the image and a dropout layer to improve training. The final output was a sigmoid categorizing the image (or assigning it score).

Generator followed similar procedure, but in reverse. We decided not to use transpose convolutional layers due to the checkerboard pattern it can introduce[8]. The network takes Gaussian noise, and lets it through a 2D upsampling layer, which performs a nearest neighbor upscaling, and then through a 2D convolutional layer, which "smudges" the information from the neighboring neurons and adds detail. The output layer uses tanh as the activation function with the appropriate value range to cover all colors. This brings the image to 128x128 pixels in size. Afterwards, this image is routed through a separate neural network which converts it to 252x252 pixels, and finally through a third one which increases the size to 500x500 pixels. All upscaling networks were trained separately. This is done to bypass hardware limitations on training large neural networks on our hardware. This image is then upscaled to 1000x1000 pixels. We consider this a reasonable approximation because 500x500 image contained enough information to achieve a remarkable score on a Kaggle testing set. Image pixels are normalized in the range [-1, 1], and then converted to frequency domain. Normalization of FFT data by subtracting the mean and dividing by the standard deviation also brings a small improvement during the training. We also experimented with Haar wavelets due to the locality of stars in the images. This data is then fed to the discriminator neural network which consisted of 3 convolutional layers. Kernel sizes in layers are small (3 to 5). Dense layer size has 256 neurons. By changing the values of parameters we determined that kernel size and dense layer size do not affect the result very much. Adding additional layers and making the network "deeper" leads to an increase in precision, even though its effect was rather small - 5% improvement within the metric in the 2 layer case. This is to be expected because local information from previous convolutional layers is further processed in subsequent layers. However, training a deep neural network requires large hardware resources, which limits us to only 4 convolutional layers.

Generator input noise is selected as Gaussian to speed up the training and give better results. The only post-processing was converting output of the network to the colors, and resizing the results to required resolution. For the training of the discriminator we use all images labeled as "good" from the labeled dataset, as well as all images with a score larger than 4.0 from the scored dataset. Unfortunately, due to limited resources, images are downscaled to the resolution of 512x512 pixels. Parameters have been tuned by training it on the provided datasets - scored for the rating network and labeled and scored (with high scores) for the GAN pair. Rating network is evaluated on the Kaggle dataset which enabled us to provide 5 submissions a day. We consider this the best practice because it gives us an objective measure of progress, and enables us not to withhold any data from the

(a) Real cosmological image (b) Fake cosmological image (c) Bad cosmological image labeled as fake (d) Spectrum plot of good image (abs. logscale) (e) Spectrum plot of bad image (abs. logscale)

network during training. For the rating network both MSE (mean squared error) and MAE (mean absolute error) give adequate results with the FFT data. However, when we try to train the network with spatial data, MAE makes the network more prone to collapse at a local minimum. With FFT data collapses during training were rare. After training, we used Kaggle to make sure that MSE convergence is close to MAE metric used on the website. For the image generation we had to use a classifier, so neither MSE, nor MAE, provide adequate loss measurement. Instead, binary crossentropy is used, which is a standard metric in classification problems. In both cases ADAM is used to find the minimum of the loss function. We use numpy and scipy for numerical processing of the images (eg. FFT), and PIL for image manipulation. Keras Framework is used to build the neural networks[7].

IV. Results

To compare our approach to the state-of-the-art models, we implement two baseline algorithms and measure the mean squared error loss during optimization, as well as the mean absolute error calculated by Kaggle for a final prediction. For the first baseline algorithm, we use the discriminator proposed by Goodfellow at. al [2], which is basically a neural network with 3 dense layers and sigmoid activation function. The second baseline is the discriminator of the deep convolutional GAN proposed by Alec Radford and Luke Metz [3]. Their model structure has four convolutional layers combined with a Leaky ReLu function and dropout layers. This part covers the discriminator of our system.

For the generator, we do not have a measurement function to evaluate the quality of the generated images. Therefore we visually evaluate the quality and compute the Frobenius norm between a generated image I_g and all the train images X_t to make sure that the network was able to learn the real distribution. If $\forall I_i \in X_t:\ \operatorname{argmin}_i ||I_g - I_i||_F^2$ is exactly zero, our generator just learned one image and not the real star distribution.

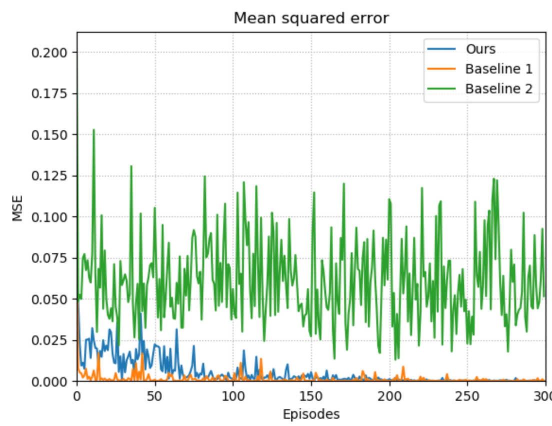

Figure 1. Our method compared with the two baseline methods

In figure IV we can see that the baseline 2 (DCGAN) is not converging in 300 episodes and pedals around 0.07. On the other hand, the baseline 1 converges rather fast. Our implementation needs around 150 episodes to converge to a mean squared error of 0.0001. However, on the validation dataset provided from kaggle, we can see that baseline 1 clearly outperform baseline 2, with nearly half the mean absolute error. In table I we can see that our method outperforms both baselines and has a mean absoute error of 0.18, which is 4 times lower than the baseline 1.

Table I
Evaluation on the test set from Kaggle

Method	**Prediction error (MAE)**
Ours	0.18157
NN (Baseline 1)	0.89831
DCGAN (Baseline 2)	1.71341

The first stage of a generator provides 128x128 images of variable visual quality. This is due to the locality of stars. It may happen that many pixels are activated with low intensity, instead of only a few with high. This leads to the appearance of light patches which could be interpreted as nebulae. However, these patches don't appear in the training set and are artifacts of the generation in the spatial domain.

This could be one of the reasons why rating of images in the frequency domain gives better results - larger patches appear as completely different frequencies, no matter where they are in the image.
Feeding these artifacts to the subsequent neural networks actually magnifies them, because they were trained to upsample cosmological photos. This makes detection of lower-quality images trivial. After the second upscaler, they will be converted to a sequence of random vertical stripes. If the first network generates a high-quality image, the output of the final image is of high visual quality and cannot be distinguished from the original images.

V. Discussion

Scoring data with randomly distributed salient local features with a neural network can lead to frequent training collapse when done in the spatial domain. However, when converted to the frequency domain, we seem to get more stable training, and better results. This could be because of the Fourier transform of the image does not depend strongly on the exact position of the star, but on their number, intensity and frequency in the image. Our experimentation with Haar wavelets instead of FFT failed to produce any useful results, and is comparable to the spatial domain, with higher resource usage.
The convolutional neural network seems to be a good fit for the problem due to the locality of the features we are looking for. In the spatial case, networks would need to be deeper to correlate data from distant areas of the image. Fortunately, FFT does this for us automatically and lets us get away with shallower networks.
Generation in the spatial domain frequently activates unnecessary pixels, which tend to confuse upscaling networks. Thankfully, results of upscaling of these images are quite easy to detect due to artifacts introduced, so this interaction of multiple networks can be ignored.

A. Challenges and limitations

One of our major challenges was it to connected our discriminator with the generator. Since our discriminator is operating in frequency space, but we need the results to be in the spatial domain, we need to compute the inverse Fourier transform at some point. To tackle this challenges we tried various combinations, like generating images directly in frequency space and feeding them to the discriminator or generating image in the spatial domain and transforming them into the frequency domain. The main issue for the first approach was that minor errors in the frequency domain result in large errors in the spatial domain. For the second approach, the framework we used did not provide a way to compute the back-propagation with the Fourier transform layer, between the generator and discriminator. Further research should be conducted in coupled frequency-spatial domain GANs.

Another issue with the data set is the size of the images. Every image is 1000x1000 pixels with one channel taking roughly 1 Mb per image where each pixel has 8 bit. Since our scored dataset consists of 9600 images, for each image we compute the Fourier transform, which outputs one complex128 array, resulting into a memory usage of roughly 153.6 GB. For that reason we need to find a way to reduce the amount of data used to train our method. During our research we tried various methods, from operating on tiles to random patches of different sizes. Finally, we found out that downsampling with the Lanczos resampling to the size 512 times 512 gives the best results. Note that even with this, we managed to obtain adequate results, but at a price of training the network on batches of 8 images.

Even though wavelets should detect local changes better than the Fourier transform, we failed to notice any improvement over the spatial domain scoring. This could be because our problem depends both on local (shape and brightness of a star) and global data (their relative position and number).

We also tried generating patches independently and discusses magnifying patches of a smaller image separately. However, these methods lead to weakly correlated neighboring parts of the image, and poor visual characteristics.

VI. Summary

In this paper, we managed to present a novel technique of scoring images with strong local features which should be positionally independent. We also pursued several directions in generating such images, and presented a technique which works in conditions of limited hardware resources. We also identified several techniques which don't provide an improvement over the current state of the field, (Haar wavelets, tiles and patches) and a couple of others which warrant more research (complex neural networks and spatial-frequency domain GANs).

VII. Further work

We discovered that the scoring network used for Kaggle submission gives the best result when given complex number frequency domain input. This is peculiar and further research in the direction of complex neural networks is needed.
Another interesting direction is to make the GAN pair operate in different domains. Discriminator judges the image better in a frequency domain (evident by Kaggle results), while generator generates an image better in a spatial domain (visual inspection). Progress in this area would lead to higher quality generated images, with fewer artefacts, and would probably eliminate ”the nebulae” which we see on spatial GANs.

REFERENCES

[1] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He, "Detectron," https://github.com/facebookresearch/detectron, 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 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[3] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *CoRR*, vol. abs/1511.06434, 2015. [Online]. Available: http://arxiv.org/abs/1511.06434

[4] K. Schawinski, C. Zhang, H. Zhang, L. Fowler, and G. K. Santhanam, "Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit," *CoRR*, vol. abs/1702.00403, 2017.

[5] K. R. Rao, D. N. Kim, and J.-J. Hwang, "Fast fourier transform - algorithms and applications." Springer Publishing Company, Incorporated, 2010.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

[7] Keras: The python deep learning library. [Online]. Available: https://keras.io/

[8] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, vol. 1, no. 10, p. e3, 2016.