Neural Networks

Prof. W. Kowalczyk

Assignment Report

Assignment 3

1 Introduction

We report on the use of a Generative Adversarial Network (GAN) to generate images of galaxies. For this, we used data from two different sources: the Sloan Digital Sky Survey (SDSS) and the Hubble Space Telescope.

Fantasy, artistic creativity has always been one of the most important skills of humans upon which we discern Homo Sapiens Sapiens from the rest of nature. This ability to create something entirely new merely from elemental building blocks is something we think of as genuinely, and exclusively, human. In the recent years, a certain kind of Neural Network has emerged and dethroned our hybris to think that creativity could not possibly emerge from somewhere else other than our human minds or the universe. So-called Generative Adversarial Networks (GANs) are able to paint and draw, compose and dream, anything from cars, human faces, bed rooms, galaxies using nothing more than the graphics card in your computer.

For this project, we will train such a GAN on thousands of real-life observations from two of the most important astronomical observatories: the Sloan Digital Sky Survey, and the infamous Hubble Space Telescope. We then let the network play divine creator, and let it produce images, observations of galaxies based upon the observations it has seen before. The reason why we are using two different sources is that we want to compare the output of the network when fed two very different kinds of pictures of galaxies.

In the following section we will begin with describing our two data sets. Here, we will characterize important feature of the images that will became important later, and show a selection of actual images for each of the two data sets. We will also elaborate on what we expect from as well as our hopes regarding he performance of the GAN. We proceed with a detailed general explanation of a GAN and its workings. After that, we separately discuss the output of the GAN for both data sets. We will also again take up our earlier thoughts and expectations of the networks performance. We conclude with a summary of our aims and the setup of our project as well as the results of the GAN and end this report with ideas how to continue on and expand our work.

2 Description of the Data

2.1 Data from the SDSS Catalog

The Sloan Digital Sky Survey was an optical survey with the SDSS 2.5M telescope at Apache Point Observatory, New Mexico, USA. Over the course of its life, the survey collected photometric data of 14.555 square degrees of the sky, thereby cataloging 930.000.000 objects and obtaining spectra for about 1.5 million galaxies (as of data release 9). The SDSS data (available from the Kaggle website: [**SDSS-data**]) is divided into a training set encompassing 61.500 images (800 MB) and a test set with a total of 80.000 images (1.01 GB). These colored .jpg-files all have a size of 424x424 pixels. Due to insufficient memory and not needing all of the data, we restricted our data to 1.000 pictures from the test set. Figure 3 gives an overview of the shapes and artifacts present in the data.

As can be seen, the galaxies span a variety of forms and shapes. Some of them show a elliptical, elongated form often with a bright bulge or spot in the middle, the dense galactic core (lower

Table 1: Table of six galaxies from the SDSS catalog.

left and lower right). Despite the rather bad resolution, a spiral structure can be recognized in a few galaxies (slightly in upper left, strongly in upper center). Other galaxies do not show a bright bulge or exhibit significant substructure and instead just have a homogeneous color (upper right). We note that nearly all pictures show foreground stars near the galaxy, ranging from just one or two to dozens of stars in the snippet. The size and accordingly the brightness of these stars range from dim spots that only cover a few pixels to very bright stars with a diameter of 50 pixels and more, sometimes even displaying diffraction spikes (upper center and strongly in lower left). Also, there are some pictures which display not a isolated galaxy but a pair or even a cluster of them (lower center and lower right).

We expect that the generated galaxies will mostly be whitish blobs throughout, often lacking any substructure. Furthermore, the generated images will likely show a number of stars of varying brightness and size. These stars however will probably be smaller rather than bigger since smaller stars appear much more frequent in the training data than larger, brighter stars. Also we hope that at least some of the galaxies will exhibit a spiral substructure. Although this feature rarely strongly appears in the data, it is frequent enough that the generator should pick it up in the later stages of the training. Lastly we hope that the generated galaxies will show features than just a featureless form. Ideally a few of them would have elongated, symmetric shapes of varying ellipticity and orientation, or would even show substructures like spiral arms, bright cores and distinguishable halos.

2.2 Data from the Hubble Space Telescope

In contrast to the former data, the infamous Hubble Space Telescope (HST) observed with a much higher magnification. This results in a much more detailed picture but also a much smaller field of view. To put in simple words: While the SDSS shows a galaxy with few details but includes its surrounding area, the HST gives a close-up view of the galaxy. However, a consequence of the latter is that the pictures might show only the central regions of the galaxy thus dismissing its outer region. The neural network is likely to pick up on this feature. In this case, the generated picture will show little to no resemblance to how most people visualize a galaxy. The data set encompasses 91 pictures whose sizes vary from less than 100 to 400 pixels from this website: [**Hubble-data**]. Table 2 shows a selection of nine galaxies out of the Hubble data set to showcase the variety of features in the data. We note the possible case of a galaxy appearing in both data sets.

The figures show a variety of colors, forms, and types of the galaxies. Some appear edge-on (upper row) thus showing the extend of the stars and their halos, as well as the dust present in the galactic plane, in a stark contrast to the bright body of the galaxy. Other were observed face-on rendering visible their dust lanes and star forming regions. However: the majority of the observed galaxies are spiral galaxies. Thus we expect to see some spiral substructure in at least some generated renditions. Another frequent feature is the bright galactic core, visible in all galaxies in the figure but best in those in the upper row. We expect that the generated galaxies will also have a bright galactic core, at least in later stages of the training. Predominant colors in the pictures are large regions of a white/yellow hue (upper row), thin regions of red, and arcs of blue (both visible in the center galaxy and the lower row in the figure). The thin, spider-like, red or brown structures in the galaxy show dust that obscures the structures behind them whereas the extended white/yellow “glow” around the galaxies resemble the actual distribution of stars and their halo. The blue/purple arcs and isolated bright blue blobs in the spiral arms on the other hand are regions of ongoing star formation. We expect the generated galaxies to show all of the three features above, especially the white/yellow glow. As it is the case in the data from the SDSS, there are numerous foreground stars visible in pictures. Again, they are likely to appear in the generated pictures.

Before we train the network with this data however, we emphasize an important caveat. The data set is exceptionally small with merely 91 pictures. A neural network, of any kind for that matter, needs a sufficiently large data set whose elements cover a sufficiently large variety of properties for it to produce meaningful - and in our case “new and original”! - content. A small data set introduces the danger of over fitting especially when paired with a wide variety of properties, i.e. when the input (e.g. the training pictures) varies too much, and this danger rises when the learning process continues. When the number of epochs is orders of magnitudes larger than the size of the input, the network doesn’t learn more but becomes used to the data it has already seen a number of times. In the case of our GAN, it won’t paint new galaxies but will try to reproduce the input data. The longer the learning, the closer the generated renditions will be to the learning data. Thus, we expect the generated galaxies to look similar as the original Hubble observations, especially for the later stages of the learning process.

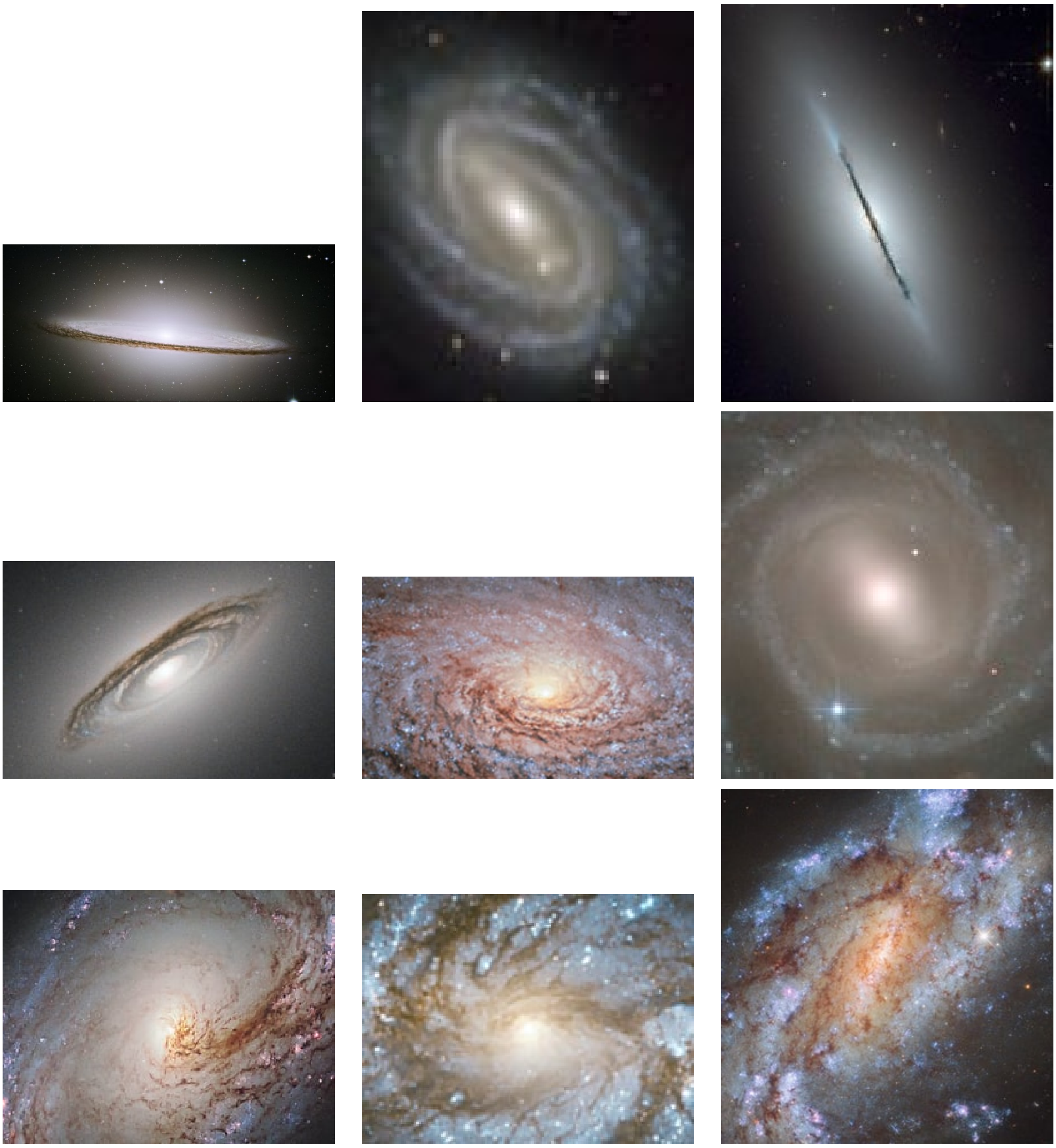


Table 2: Table of selected pictures in the original Hubble data set. They give an overview of the predominant features of the Hubble data like the high frequency of occurrence of spiral structures, dust lanes (especially upper row), as well as the color distributions.

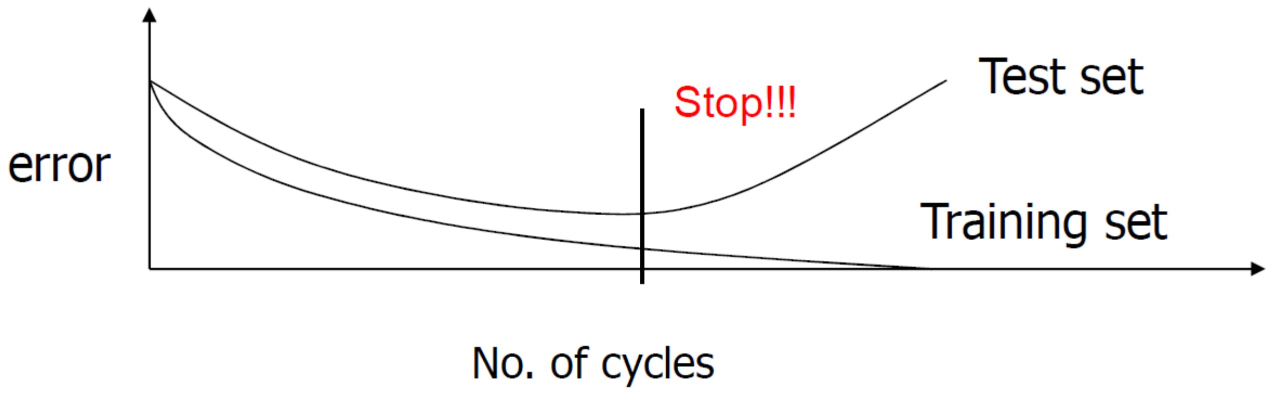


Figure 1: The general error of a Neural Network at a respective epoch during the training process. Naturally, the error frequency is lower on the training than on the test set and decreases as the learning continues. At a certain point however the network gets used to the training data too much (black vertical). Further training will further reduce the error frequency on the training set, but will rapidly increase the frequency for different data. At this stage, the network is over fitted to the training set.

3 The Setup

The DCGAN (Deep Convolutional Generative Adversarial Network) consists of two different distinct neural networks: the Generator and the Discriminator. The job of the Generator is to generate images that resemble the images of the training data as closely as possible. The job of the Discriminator then is to distinguish the real images pulled from the training data, from the fake images generated by the Generator. See figure 2 for a visual representation.

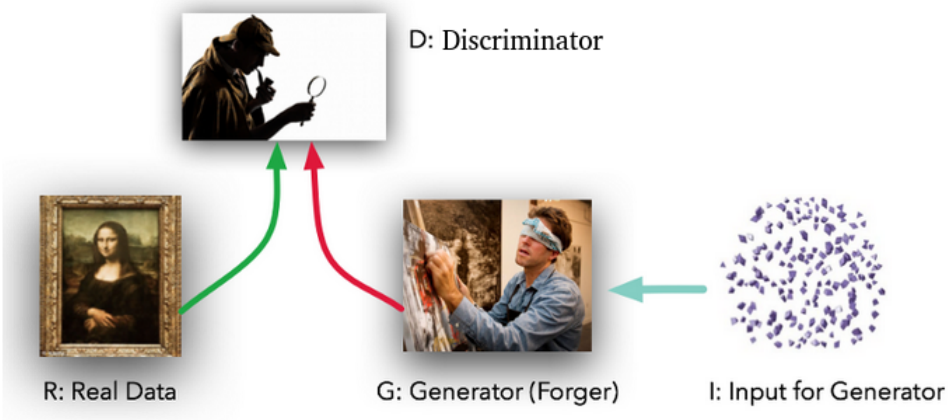


Figure 2: Schematic of the idea behind a Generative Adversarial Network.

The learning process goes as follows: the Generator is provided with a noise vector which is a vector of a certain length filled with random numbers. From this vector the Generator generates its first image that should resemble the picture type of the training data. However since its the first generated image the Generator has no clue on how to generate a believable image thus it pretty much looks like random noise. The Discriminator then takes a look at the real images as well as the generated image and labels them either as true (1) or fake (0). Since the Discriminator hasn't been trained either, at the start it will try to distinguish based

on very shallow features like for example the color of the image. This information of how the Discriminator distinguishes between real and fake is then passed back to the Generator. The Generator then uses these features in the images it generates to try and fool the Discriminator. The idea is that this process will be repeated until the Generator generates images that look so real and like the training data that the Discriminator is no longer able to distinguish between the real and the generated images.
The framework of the Discriminator looks as follows:

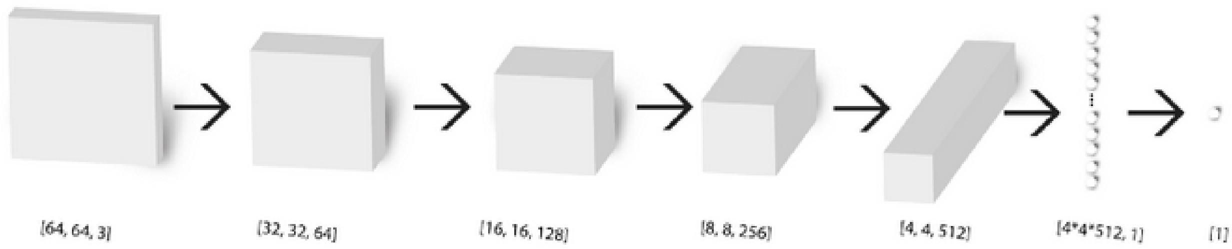


Figure 3: Discriminator framework

The layers are convolutional layers to pick up features in the images, except for the last layer which is a logit to either output a 1 for a real image or a 0 for a fake image.
The framework of the Generator looks as follows:

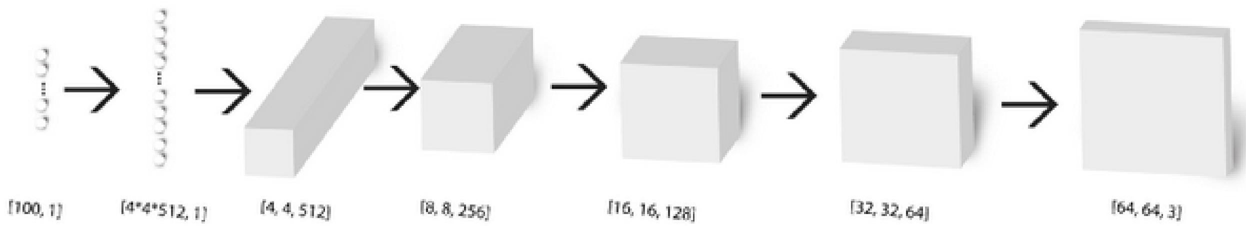


Figure 4: Generator-framework

The input vector is a noise vector consisting of 100 random decimal numbers and the subsequent layers have the same structure as the Discriminator framework however the order is reversed. So instead of a convolutional layer the layers are transpose convolutional layers that produce images given input numbers. The output layer being a 64x64x3 tensor representing the 64x64 three color generated image. The activation function used in the layers that have one is the LeakyReLU activation function, which for us is:

$$LeakyReLU(x) = 0.6x + 0.4|x| \tag{1}$$

and looks the same as the normal ReLU function for x>0, however for x<0 it still has a small gradient as opposed to the ReLU function which is 0 for x<0 and thus has a gradient of zero. The LeakyReLU function can be seen in figure 5. This non-zero gradient for x<0 prevents weights from dying when learning the network, because multiplying with a gradient of 0 causes all subsequent layers to die out. This problem is commonly known as the dying ReLU problem.

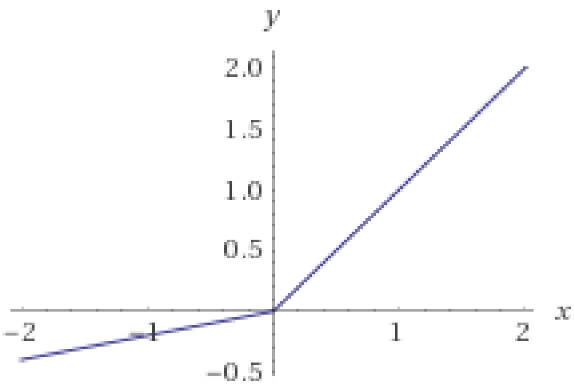
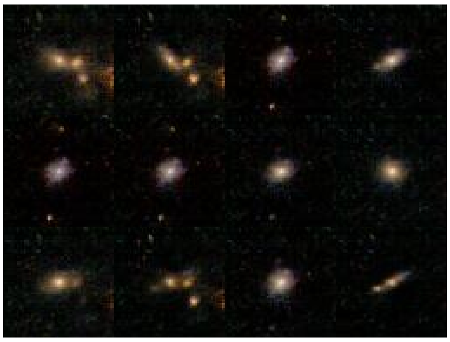


Figure 5: LeakyReLU function for -2<x<2

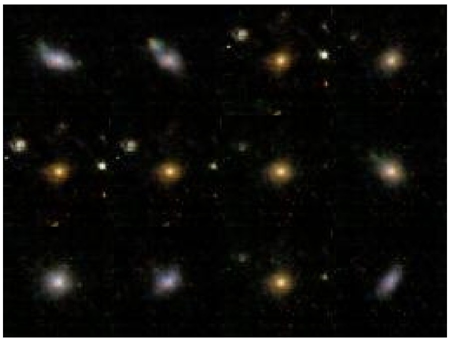
The loss function that is used for the Generator and the Discriminator is the sigmoid cross entropy with logits function that is a standard loss function in Tensorflow. We will use a batch size of 12 images, so every epoch the generator will generate 12 fake images.

4 Results from the SDSS data

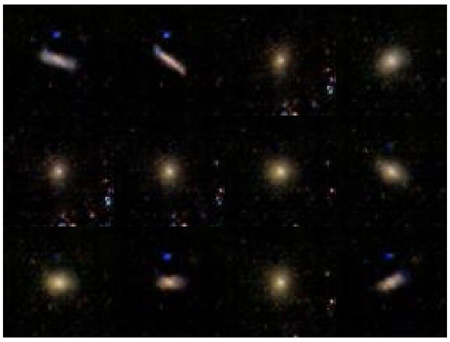
The generated images of galaxies generated after the training on the SDSS data show a variety of shapes which can be seen in the figures of table 3. Some even appear with elongated, cigar-like shapes as after epoch 28.200. The network also produced pairs of galaxies, as can be seen after epoch 27.000 and possibly 27.400, and even multiples and clusters of galaxies like after epoch 26.400. However, the network had a hard time to produce significant substructure, especially spiral arms. It only produced a few candidate galaxies where one might recognize such a pattern behind the noise with a little bit of fantasy. Regarding the missing spiral arms, there are two possible reasons. Firstly, there are not many spiral galaxies in the data, or rather we were able to discern such a structure in only few galaxies of the data set. When we who know what we were looking for in the pictures had a hard time to see a spiral structure in more than just a few galaxies, the neural network - who doesn't know about the existence of this structure - would have a much harder time to pick up on this pattern. Secondly, the feature is rather subtle. It is merely a locally constrained periodic variation of the galaxy's surface brightness with a very small contrast. It is probable that these patterns simply are dwarfed by the noise in the picture. A straightforward solution for this is to feed the network more data where this structure is more prominent. This could be achieved either by feeding it more spiral galaxies or replacing a part of the data with galaxies showing a prominent spiral structure. Lastly, the images often show a "background" of stars, that is a small number of bright point sources. This fulfills our expectations of the GAN picking up on general patterns of the pictures that we do not intended to have in our pictures.



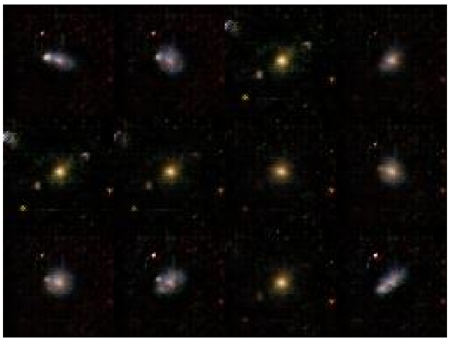
Epoch 26.400



Epoch 27.000



Epoch 28.200



Epoch 27400

Table 3: Table of generated galaxies from the SDSS data. Clockwise beginning with the upper left picture, these renditions were generated after the training epochs 26.400, 27.000, 27.400, and 28.200

The results satisfy our hopes that the network would generate galaxies with a wide variety of shapes, and even pairs and clusters of galaxies despite them not appearing often in the training data. Nonetheless, the network nearly never produced galaxies with prominent spiral arms. This is despite the fact that a (albeit small) number of galaxies in the training data share the (albeit weak) feature of a spiral structure.

If we look at the loss rates of the Generator and the Discriminator in the figures of table 4 we can see a general larger trend in the Generator loss rate steadily increasing and the Discriminator loss rate steadily decreasing with a smaller scale trend of peaks and valleys. The peaks and valleys are a result of the fact that the Generator and Discriminator are competing with one another and when one gets ahead the other gets behind. For example if the Discriminator is suddenly able to pick up a certain feature the loss rate will decrease for the Discriminator because it can use it to tell fake and real images apart better. At the same time this means that the Generator loss rate will increase, because its generated images are now suddenly being better detected as fake. The global trend of the Generator loss steadily increasing and the Discriminator loss steadily decreasing tells us that the Discriminator is 'winning' the battle and is getting better faster at distinguishing fake from real than the Generator is able to generate realer looking images.

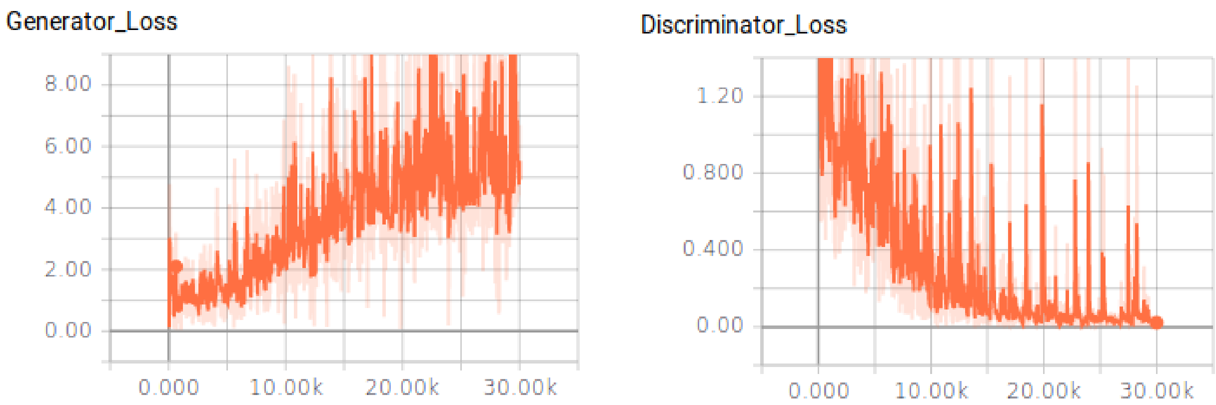


Table 4: Loss rates of the Generator and Discriminator for the SDSS data set for iterations 0 to 30k.

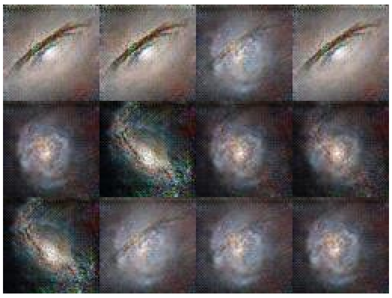
5 Results from the Hubble data

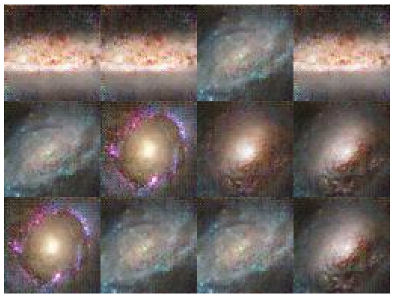
We trained the DCGAN for a total of 30.000 epochs on the Hubble galaxy data set. Table 5 shows a selection of generated galaxies after different epochs. The first images that could resemble a galaxy were produced after epoch 8400 (see upper left). The five galaxies are very similar and could maybe be mistaken for a noisy observation with a much lower resolution. Noise is present throughout the picture and especially in the fringes of the picture. The network apparently tried to paint a spiral galaxy with limited success. The generated pictures show a colorful variety once the network started to produce meaningful output after about 10.000 epochs of learning. After some time, especially after 20.000 epochs, the 12 generated renditions seem to come from a few examples and renditions with the same model show very close resemblance to each other. This could be a consequence of over-fitting.

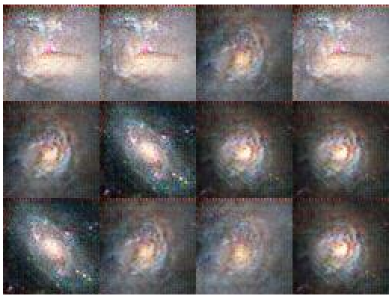
Some pictures are not a colorful depiction of a galaxy but rather an indiscernible, colorful chaos as after epoch 29.000 and 29.200. These likely were inspired by the aforementioned close ups of galaxies that only show the inner regions of a galaxy. When we fed the network pictures where the galaxy is missing a definite edge to the blackness of space, we have to expect that the network produces a colorful chaos of white, red, and yellow. However, these pictures still have their “storm” (which should have been said inner regions of a fantasy galaxy) encircled by a region of much lower brightness. Examples of this can be seen in the upper row of epoch 29.400. This means that in these cases, the networks tries to combine the output of these inner regions with a very common feature of the other renditions: a black or dark rim around the galaxy, i.e. the transition from galaxy to space. We however do not know why the network painted the “fire wall’ in epoch 29.000. This is especially an interesting case since the “fire”, lacking a better word, only is present in the upper half of the picture while the transitional mid area and the lower half look like a galaxy that was split near its core.

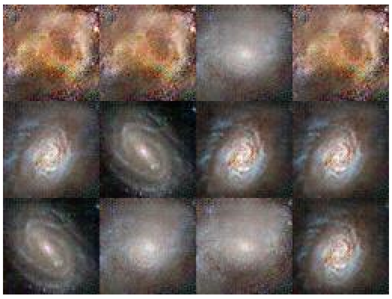
Nonetheless, most renditions feature a remarkable variety of details. These include prominent spiral structures like in epoch 29.400, the dust lanes in epochs 29.800 and 27.600, obscuring the galactic center with a strong contrast, and bright galactic cores like in epoch 27.600, 29.000, and 29.200. Sometimes these could be mistaken for actual observations as they are remarkable close to some pictures in the training data, for example in 27.600 and the two Grand Spirals in epoch 29.400. For the latter case, we show a very similar original observation in the upper center of table 2. This phenomena is almost certainly a result of over fitting due to the small size of the input data as well as the large number of training epochs, as was discussed in section 2.2.


Epoch 8.400


Epoch 27.600


Epoch 29.000


Epoch 29.200


Epoch 29.400

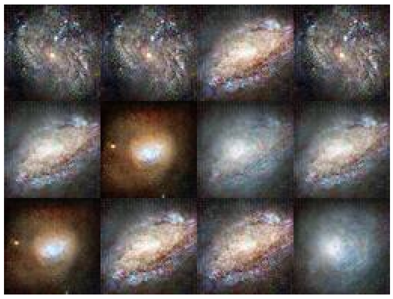
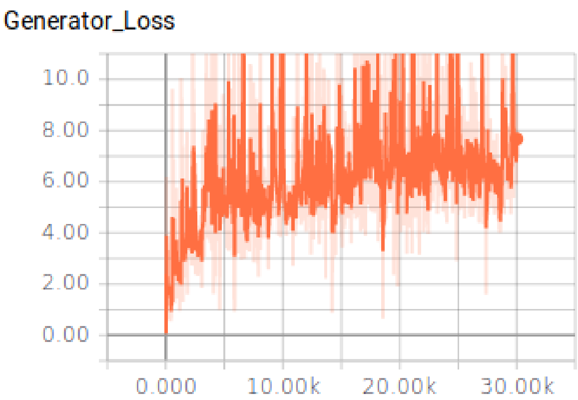

Epoch 29.800

Table 5: Table of generated galaxies inspired by the Hubble data. The images in the upper row from left to right were produced after epoch 8.400, 27.600, and 29.000 and those in the lower row were produced after epoch 29.200, 29.400, and 29.800

Looking at the figures in Table 6 we see a very similar story as we saw in the SDSS loss rates of the Generator and the Discriminator: the spikes in the loss rates signifying the competition between the Generator and the Discriminator as well as the global upward trend for the Generator loss and the global downward trend for the Discriminator loss. The only difference is that the SDSS Generator loss was increasing linearly on a global trend and the Hubble generator loss seems to have reached some kind of asymptotic limit. This could be the result of the overfitting due to the Hubble data set being one order of magnitude smaller than the SDSS data set.



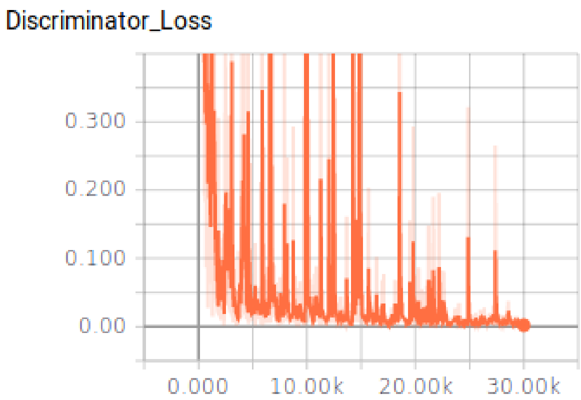


Table 6: Loss rates of the Generator and Discriminator for the Hubble data set for iterations 0 to 30k.

6 Conclusions

For this project, we have used a Generative Adversarial Network to produce pictures of galaxies. As training data, we used observations from the Hubble Space Telescope (HST) and the Sloan Digital Sky Survey. We obtained 91 observations from the archives of the HST with varying sizes between less than 100 and 400 pixels, and 1.000 recordings from the SDSS with a common size of 424x424 pixels.

The GAN performed quite well with the SDSS data. The generated pictures resemble the observations we fed the network, while representing new renditions with a variety of different features. The pictures vary in number and brightness of background stars, and the depicted galaxies vary in shape, orientation, and structure, and sometimes appear together in pairs or even clusters. However, it was not able to really pick up on and reproduce spiral structure in galaxies. Although the shape of some of the generated galaxies lead to the assumption that they might be spirals, it appears that the training data exhibits this feature (that is spiral arms in the galaxy) too subtly, or that the training data does not encompass enough observations with a prominent spiral structure. A solution for this is to use a bigger data set with relatively more prominent spiral galaxies, or to train the network for a longer time.

The performance on the Hubble data turned out to be similarly successful. The larger magnification of the telescope and the resulting smaller field of view proved to be both a blessing and a curse. Since the training images exhibit much more details, the network was able to produce renditions of galaxies that showed a variety of shapes and features like dust lanes or more and less prominent star forming regions. Some of the best renditions could pass as actual observations on the first glance. The variety of shapes and colors of the renditions was impressive. The small field of view however resulted in some pictures that only show the bright inner regions of the galaxy, without the outer dimmer regions and the transition to the darkness of space. When the network learned the pictures with the missing definite boundary of the galaxy, it produced colorful but meaningful "chaos". It seems that a stark contrast marking the boundary of the object helps a lot to produce meaningful and realistic renditions. Furthermore, some renditions often closely resembled the training images of Hubble observations. This is especially the case in the later stages of the training process. Figure 2 for example shows a galaxy (upper center) which bears a striking resemblance to some galaxies that the GAN produced after 29.400 epochs, as shown in figure 5. The reason for this is likely an over fitting of the network. The size of the training data is very small with 91 images compared to recommended sizes of training data for such networks. The effect of this over fitting can be discerned very easy in the renditions produced in the last epochs of the training, since the effect becomes stronger the longer the training continues. There are two way how to avoid this problem. Most importantly, one should feed the network a much larger data set of at least some tens of thousands of pictures. Secondly, however this is not the best strategy, one can cancel the training process much earlier before the over fitting sets in or becomes dominant. Alas, the then generated renditions of galaxies (or any other object for that matter) might be much less realistic and impressive.

There are two ways one could elaborate on this work. The most straightforward way is to use a larger training set for the network. Not only would this solve the aforementioned problem of over fitting, it would also introduce a larger variety of shapes and orientations to the network. Since the training data of Hubble mostly consisted of spiral galaxies, the output of the network encompassed more or less only variations of spirals. It would be interesting to see how the GAN works with detailed observations of a variety of Elliptic, Dwarf and Spiral Galaxies. Also, one can expand the scope of the training. There is much more out there in space than galaxies - what would happen if the GAN is trained with pictures of not only galaxies, but star clusters, and nebulae?
Where is the point that the variety of shapes and forms in the training data is simply too much for a network, and is this problem solvable with just a longer training process?
Would a network be able to discern between these objects when shown observations from a different telescope?
What if it is shown two observations from the same galaxy, one with a very high resolution and many details and one with a very poor resolution: would the network still be able to recognize them as depictions of the same galaxy?
What would happen if one would use observations of the same object, be it star cluster, galaxy, or nebula, but in two different parts of the electromagnetic spectrum?
Could a neural network learn how to discern observations of the same object but in these different spectra?

References

[1] Kaggle SDSS galaxy images: https://www.kaggle.com/c/galaxy-zoo-the-galaxy-challenge/data

[2] Hubble galaxy images: https://www.spacetelescope.org/images/archive/category/galaxies/

[3] Naresh Nagabushan, original code, https://tinyurl.com/y7rpahdw