MLP Coursework 3: Interim Report

Group 2 : s1415713, s1448320, s1401631

Abstract

We investigate Generative Adversarial Networks (GAN) in the task of unsupervised feature learning utilised in face-swapping, inspired by the recently popular algorithm referred to as *deepfakes*. We focus on the impact of image characteristics, such as lighting and background in the training set on the final transformation result. We introduce four data sets, two involving Donald Trump and Hillary Clinton, and two created using the faces of the authors. Our experiments show that our model achieves satisfactory performance when trained on invariant (*easy*) data (*Section 3*), however struggling to effectively carry out the face-swap when provided with images containing varying backgrounds, lighting conditions or distortion (*hard* data (*Section 3*)) . We plan to further implement adaptive dropout (Ba & Frey, 2013) and pixel shuffle up-sampling in our model to improve our results. We also conclude that longer training is required in order to achieve definitive findings.

1. Introduction and Motivation

Recently, the *deepfakes* algorithm sparked controversy due to its use in the production of counterfeit pornographic videos featuring faces of celebrities. Although controversial, this portrays the capabilities of the model, as the generated content was indistinguishable from authentic videos (Hern, 2018).

Despite this recent controversy, face-swapping has been a topic of research for a significant time period. It has many uses, including film and photography editing, face de-identification, image combination, as well as other broad applications in creative work.

Bitouk attempted this task as early as 2008. He generated a dataset of images sourced from the internet, then preprocessed them in order to extract faces and align them to a common coordinate system. Given an input face, potential candidate replacements were ranked by pose, resolution, image blur, colour, lighting and seam signature. Face-swapping was then performed through image processing tools (Bitouk et al., 2008).

Dale extended this concept to videos, by attempting to replace the faces of two individuals performing a similar task. This introduced further complexity to the problem, in the form of temporal alignment, changing facial expressions as well as temporal consistency. We will aim to extend on this research with our *easy* datasets described in Section 3 (Dale et al., 2011).

In the previous semester, we have focused on the MNIST/EMNIST classification task using various types of neural network architectures. This involved predicting a label (i.e. category) of the given feature of a data instance (i.e. image). This kind of approach is represented by discriminative models which try to learn boundaries between classes to classify the input data. In a probabilistic sense, discriminative models directly learn the conditional distribution $P(y|x)$ for input data x and the desired output class label y (Ng and Jordan, 2014).

Generative models, on the other hand, try to model the distribution of each individual class. Instead of predicting a label given certain features, generative models try to predict features given a certain label. This means that when supplied with training data, we can generate new samples from the same distribution (Ng and Jordan, 2014). This has been the main motivation behind the viability of generative models in the face swapping task.

Contrary to their discriminative counterparts, generative models address density estimation, making them suitable for unsupervised learning. They try to understand and explain the underlying structure of the input data without access to any labels (Karpathy et al., 2016).

However, GAN has not been used for the task of face-swapping until relatively recently. GAN networks have seen substantial growth since 2014 in the task of mapping images from noise input (Dong et al.,2017). Dong defines the problem of "image to image translation" as an automated transformation of an image from its original form to a synthetic form, whilst maintaining the structure and semantics of the original image. His model utilises deep convolutional neural networks and conditional GANs. Two-step unsupervised learning is used in order to translate images between domains without labelling the images, thus avoiding the cost of labelled data. Dong's model achieves significant generality and is able to translate images with large variance (Dong et al.,2017).

We introduce the notion of *hard* and *easy* datasets, which have varying training difficulty, as described in *Section 3*. Our baseline experiments constitute training our model on these datasets, in order to understand its performance under varying training difficulty and the impact of features such as background and lighting on the efficiency of the model. We will then compare our results with those seen in the literature. For our further experiments, we aim to extend our model through the addition of adaptive dropout (Ba & Frey, 2013), as well as batch normalization, as suggested by Redford and Metz (Redford & Metz, 2016).

We will also investigate other suggestions for improving the performance of GAN networks found in literature, such as the Pixel-Shuffle (Shi et al., 2016) and more frequent discriminator updates (Chintala et al., 2016) (Gulrajani, 2017). Finally, we will produce a video of the Donald Trump - Hillary Clinton Presidential Debate with swapped faces and compare our results to those achieved by Dale on a similar task (Dale et al., 2017).

2. Research Questions and Objectives

The aims of the project can be classified into two discrete subsections. Firstly, we will investigate the Generative Adversarial Network (GAN) model (GitHub - shaoanlu, 2018) which will be used throughout our baseline experiments. Next, we will attempt to improve on the results obtained by modifying various components of the model, as described in *Section 2.1.*

We will investigate the performance of the original and modified models on datasets with a varying degree of difficulty. We introduce two types of datasets, referred to as *hard* and *easy*, further described in *Section 3.* We will analyse the impact of aspects such as lighting, background, pose, varying age and genders of subjects and the presence of foreign objects in images. This is further discussed in *Section 2.2.*

2.1. Model Architecture Research Questions

The following research questions will be addressed:

1. To what extent does the model suffer from overfitting? Can dropout be used in order to reduce the degree of overfitting? In particular, adaptive dropout (*described in Section 4.2*) will be implemented in order to investigate this further (Ba and Frey, 2013). We were unable to find any literature describing the use of adaptive dropout in a GAN network, therefore this will be novel research.

2. To what extent is batch normalisation (Ioffe & Szegedy, 2015) able to stabilise the training process of the generator and improve the overall efficiency of GAN? We are planning to apply a batch normalisation layer on all the layers in the generator except its output layer as proposed by Redford and Metz (Redford & Metz, 2016).

3. Would increasing the input dimensions (128 x 128 or more) significantly affect the length of the training process? To what extent would it improve the quality of the actual face swap? Our baseline's model input size is currently 64 x 64 which is a compromise between the quality of generated images and computation resources. We would like to increase the input size of our model

and investigate the quality of generated swaps. Moreover, this would allow us to deepen our network and extract more complicated representations (i.e. features) from the images.

4. To what extent would the sub-pixel method also known as Pixel Shuffle (Shi, et. al. 2016) improve the quality of up-sampling and how would it influence the overall model performance? We are planning to follow current recommendations regarding GAN training (Chintala, 2017, Chintala et al., 2016) to improve the quality of the face swaps.

5. Would updating the discriminator more frequently than the generator benefit our model? Recent studies suggest that training the discriminator for more steps before updating the generator is beneficial for the overall performance of GANs (Gulrajani, 2017, Chintala et al., 2016).

2.2. Data Set Research Questions

The following research questions will be addressed:

1. How does the model perform when trained with easy data? Are viable results achieved in the default configuration, and therefore can the model be used as a baseline for further experiments or does it require changes?

2. What is the performance of the default model when *hard* data sets are used for training? Can satisfactory results still be achieved when trained on images with varying backgrounds and lighting conditions? To what degree does the presence of foreign objects inhibit the training process? Does the age and gender of the subjects have an effect on the accuracy of training? Ultimately, we aim to produce a video of the Donald Trump - Hillary Clinton 2016 presidential debate (NBC News, 2016) with swapped faces of both individuals which are indistinguishable by the naked eye.

3. Dataset and Task

The experiments make use of four distinct data sets. The first two datasets consist of a series of images of Donald Trump and Hillary Clinton (Jansenh.stackstorage.com, 2017). Datasets C and D are composed of images of Albert and Karol, two members of the group.

3.1. Dataset A - Hard Donald Trump and Hillary Clinton

This dataset consists of 377 images of Donald Trump and 404 images of Hillary Clinton. The images are all cropped to a size of 256 x 256 pixels and focused on the face of the individual, eliminating background to the greatest extent possible. Although both sets of images contain data from multiple sources, it is worth noting that the Hillary Clinton set contains 255 images which are frames originating from the same video. In this sense, the Donald Trump set is significantly more diverse, as it is composed of images from a wider array of sources. This is likely to lead to a greater degree of overfitting on the Hillary Clinton training. This will be taken into account when analysing the results of the experiments on this training set. Both datasets contain images of the individuals from various stages of their lives, with differing lighting, backgrounds, angles and facial expressions. Grayscale images are also included. A number of pictures include foreign objects like microphones, hats, watermarks and flags. The dataset is therefore much noisier and should be significantly harder for the model to train on, when compared to dataset B described below. This data set is sourced from an online repository (Jansenh.stackstorage.com, 2017).

3.2. Dataset B - Easy Donald Trump and Hillary Clinton

This dataset is sourced directly from a recording of the 2016 USA Presidential Debate (NBC News, 2016). The Donald Trump set is obtained from an 81-second clip, at 30 frames per second, resulting in 2437 images. The Hillary Clinton images are retrieved from a 72-second fragment, sourced at the same frame rate and amount to 2151 images. All images are cropped to a size of 300 x 300 pixels and centred on the face of the individual. This dataset is normalised to a much greater degree than dataset A. The background and lighting in all images is relatively consistent with small deviations occurring when the individual walks across the stage. Facial expressions are limited to neutral expressions and mouth movement associated with speech. The clips were chosen such that all images are filmed from the same angle with a single camera, and the only foreign object present is a microphone. The model trained with this dataset is likely to suffer from considerable overfitting due to the aforementioned features of the data, nevertheless, it will be used in order to understand the capability of the model in close to ideal conditions. It will also be useful when investigating the impact of adaptive dropout on the reduction of overfitting.

3.3. Tasks for Dataset A and Dataset B

1. In the baseline experiments, the default model will be trained on both data sets. This will allow us to investigate how well the model trains on arbitrary images and how susceptible it is to changes in background, lighting, age differences and other aforementioned factors which are likely to inhibit the training process. It will also enable us to understand the performance of the model under relatively easy training conditions.

2. We will use the model trained on Data Set B in order to perform a face-swap on arbitrary video clips of Donald Trump and Hillary Clinton. We will also perform this face-swap with the model trained on Data Set A. This will allow us to assess the degree of overfitting in the model trained on Data Set B, and how this impacted the final result of the face-swap. We will understand whether or not this model can be viably trained using a single video of two individuals. This objective will be carried out in our further research for courserwork 4.

3. We will reproduce the above two experiments using the edited model described in *Section 2.1* in order to understand the impact of our changes on the performance of the model. This will also be done in coursework 4.

3.4. Dataset C - Hard Karol and Albert

This dataset was created by two of the group members in order to provide images which can facilitate the same degree of training difficulty provided by dataset A. In order to do so, images were sourced from various social media accounts. They include differing backgrounds, lighting conditions, angles as well as multiple facial expressions. Some images also contain Snapchat and Facebook filters in order to further increase the difficulty of training. The dataset contains 46 images of Albert and 48 images of Karol. The images are normalised to a size of 300 x 300 pixels and they have been cropped in a way which places the face of the individual at the centre of the image. Unfortunately, we were unable to source a pool of images which is the same size as that of dataset A. This should, however, further increase the difficulty of training and allow us to elaborate the capabilities of our models on a limited size training set.

3.5. Dataset D - Easy Karol and Albert

This dataset was created with the aim of maximising the consistency of the data and removing any potential factors which will inhibit the training process. The dataset consists of still images taken from a recording of both group members reading a news article. Albert and Karol were chosen due to similar ethnicity, skin colour and facial features. Both videos are recorded using the same camera (1080p HD resolution, 7MP, $\int$ 2.2 aperture), and are taken from the same angle, ensuring that the lighting, background and film angle is consistent across all images. Finally, the images are cropped to a size of 300 x 300 pixels, with the face centred in each image. The data set consists of 906 images of Albert and 875 images of Karol.

3.6. Task for Dataset C and Dataset D

1. How does the model handle image distortion such as Snapchat and Facebook filters, as well as grayscale images? Can the model be effectively trained on a small sample of 50 images? This will be assessed by training the model using both datasets and then attempting a face-swap on the video used to source DataSet D from. This will allow us to assess the behaviour of the model in close to ideal conditions - when trained on Data Set D and tested on the video used to obtain the training set from. This will be compared to the performance achieved when training is performed on a limited number of images exhibiting heavy variance (Data

Set C). These experiments will be repeated with the edited model described in *Section 2.1* in order to understand the impact of our changes on the performance achieved by the model. Although we will perform the baseline training experiments now, the video generation is reserved for coursework 4.

3.7. Task Evaluation

The task will be evaluated through a visual comparison of the images produced by the model after the face-swap. We recognise that a better tool is needed for this task. We aim to train a linear classifier on authentic images and allow it to classify our generated images into two classes - real or fake. This will behave similarly to the discriminator. We will then use the percentage of stills classified as authentic as a metric of success. A similar model has been utilised in previous research for feature detection and we believe it is applicable to our use case (Ba & Frey, 2013).

4. Methodology

4.1. Default Model

An example of a generative model is the generative adversarial network (GAN) (Goodfellow, 2014). GAN provides an attractive alternative to maximum likelihood techniques used for image generation such as PixelRNN, PixelCNN (Radford, 2015) or Variational Autoencoders (Doerch, 2016); which tend to produce blurry results (Karpathy, 2015). In contrast to the mentioned models, GAN does not work with any explicit density function, instead taking a game-theoretic approach which enables it to learn how to generate samples from training distributions through the "2-players game" (hence the name "adversarial") (Karpathy, 2015). Recent studies indicate very broad applications of GANs in tasks such as image generation (Bengio, 2016), or image super-resolution (Ledig, 2017).

The input to the network is a random noise vector which we hope to transform into a sample from the training distribution. The "2 players" are the generator and discriminator. The generator's aim is to fool the discriminator by generating fake images from the desired distribution which cannot be distinguished from their real counterparts. The generator is trained to minimise the probability of assigning a "fake" label by the discriminator. The discriminator, on the other hand, tries to distinguish between real and fake images. It is a classifier which classifies images into two classes, real and fake (Dong et al., 2017).

The general overview of the GAN training process can be divided into the two following stages:

1. A vector of random noise z is fed into the generator which outputs an image based on the training data distribution.
2. The outputted images along with the "real" one from the training set are then fed into the discriminator, which tries to determine if the generated image is, in fact, fake (assigned value close to 0) or real (assigned value close to 1).

These stages also indicate two feedback loops where the generator receives feedback from the discriminator regarding the "authenticity" of the produced samples as well as the discriminator validating its classification by comparing the distribution of the image with the distribution of the images in the training set (Chintala, Radford & Metz, 2016).

The biggest disadvantage of GAN is its instability during the training process. This makes it a constantly developing area of research, with further room for improvements (Goodfellow, 2016). The network relies on an equilibrial influence of the discriminator and generator. A discriminator which is too "tolerant" and accepts all or most of the images created by the generator would make the whole network infeasible. Similarly, a discriminator which rejects all images would inhibit the training of the generator (Thewlis, Bilen & Vedaldi, 2017). We are fully aware of the challenges regarding this aspect and would like to address and understand these issues further to improve our final results throughout our project.

4.1.1. Cycle GAN

Our baseline model is a GAN network organised in a cyclic manner and previously used in unpaired image-to-image translation (Zhu, 2017). In this task, we aim to translate an image from domain D_X to domain D_Y without the need for one-to-one mappings between the input and target domains, as required by the non-cyclic GAN model. No need for such a mapping makes the cycle network very powerful making the training process cheaper and faster due to much easier dataset extraction. Furthermore, the lack of one-to-one mappings makes the model applicable in various domains such as swapping between apples and oranges (Zhu, 2017).

The goal, in this case, is to learn a mapping $G : X \rightarrow Y$ such that the distribution of images from $G(X)$ is as similar to the distribution Y as possible. This is achieved using an adversarial loss. Due to the mapping being highly under-constrained, it is coupled with an inverse $F : Y \rightarrow X$ introducing a cycle consistency loss to enforce $F(G(X)) \approx X$ and $G(F(X)) \approx Y$.

Our approach is very similar to standard GAN except we have separate generators and discriminators for each of the domains (person's A and person's B faces). An input image from domain D_A is fed into the first generator which transforms the image into another image in the domain D_B. The resulting image is then fed to another generator which transforms it back to the image in domain A (hence the cycle) (Hoffman et al., 2018). The final image has to be as close to the original image from domain D_A as possible in order to achieve a meaningful mapping which has to be learned due to the lack of a paired dataset. In the next step, pairs of images from each domain are fed into the appropriate discriminator which tries to distinguish between these pairs and either reject or accept the generated image depending on the quality of the generated counterfeit image (Hoffman et al., 2018).

4.1.2. Generator

Our model consists of two generators and two discriminators, one for each class. Each of the generators (i.e. auto-encoders) share the same decoder. Our baseline model builds upon Deep Convolutional GANs (Redford, 2015) which replace any pooling layers with striped convolutions in the discriminator and fractional-strided convolutions in the generator. This yields much better results by allowing the network to learn its own up-sampling and down-sampling.

In our approach the generator is a fully convolutional auto-encoder consisting of a collection of convolutional and deconvolutional layers, to reconstruct the original unmasked image. The discriminator is then trained using the output of the generator. The discriminator's output is used together with a reconstruction cost to update the weights of the generator (Chintala, Radford and Metz, 2016).

Due to a fixed 64 x 64 RGB image input size to each generator, all the images need to be resized to fit the appropriate dimensions. The first step in the generator is encoding, during which the convolutional network (encoder) tries to extract the features from the input image. The encoder consists of four blocks of convolutional layers with a kernel size of 3, stride 2 and the ReLU activation. Each convolutional block leads to the extraction of more higher level features. It can also be seen as down-sampling the image into 1024 features maps of size 4x4 each. These features are used as the input to the decoder in order to transform a vector of features in one domain to the feature vector in the desired domain (Thewlis, Bilen and Vedaldi, 2017).

After feature extraction from the images of both classes by the shared encoder is completed, two separate decoders are used to reconstruct the original image using a loss function that trains on minimising the loss between an image and its warped variant (as described in *Section 4.1.4*). In this operation, transposed convolutional layers are used in order to transform low-level features back from the feature vector. Our decoder consists of 4 transposed convolutional blocks with a kernel size of 3 and stride 2. Leaky ReLU has been used as an activation function.

Therefore, given images for person A and person B, we train the auto-encoder G_A on person's A images and G_B on person's B images. To generate fake images, we are feeding a representation of A to the decoder in G_B and vice versa for the reverse operation (i.e. swap person's B face into person's A).

4.1.3. Discriminator

The role of the discriminator is to distinguish whether the input image is a real face. The input to our discriminator is an image of size 64 x 64 which has been chosen as our default image size due to the training time constraints.

In order to extract the features from the input image, our discriminator consists of 3 convolutional block layers each with a kernel size of 4 and stride 2, reducing the dimensions by a factor of 2. These strided convolutional layers tend to yield better results than traditional max unpooling layers allowing the network to learn its own downsampling (Radford, 2015). Moreover, each convolutional block also contains the LeakyReLU activation function with an α parameter of 0.2 which has been chosen due to recommendations in recent research (Radford, 2015). To allow the discriminator to distinguish between fake and real images, the final convolutional layer which produces a 1-dimensional output has been added and a sigmoid function is used on that layer to yield a probability between 0 and 1. The discriminator is rewarded for assigning a low probability to generated faces, and a high probability to real faces. The cost functions are described below.

4.1.4. Loss function

In standard GAN networks, the discriminator is treated as a classifier with a sigmoid cross entropy function and is trained simultaneously with the generator. As previously described, the generator aims to learn the distribution p_g over training data x by sampling a random noise vector z from a uniform distribution $p_z(z)$. Then, using a differentiable network, the input is mapped to data space $G(z; \theta_g)$. The discriminator is in fact a classifier $D(x; \theta_d)$ that tries to distinguish between real and fake data. Hence, the minimax objective for GAN is formulated as seen in *Equation 1*:

$$\begin{gathered}\min_G \max_D V_{GAN}(D,G) = \qquad (1)\\ \mathbb{E}_{x\sim p_{data}(x)}[logD(\mathbf{x})] + \mathbb{E}_{z\sim p_z(z)}[log(1-D(G(\mathbf{z})))]\end{gathered}$$

However, recent studies found that this loss function may lead to the vanishing gradients problem during the learning process (Metz, 2016). One of the approaches to handle this issue is LSGAN (Mao, 2016). LSGAN is a variation of original GAN which uses least squares loss instead of cross entropy loss. The main issue with cross entropy loss is that when updating the generator, the datapoints far away from the decision boundary, which are still on the "positive" side of it will be classified the same way as the points being closer to the boundary. This is because the cross entropy loss does not take into account the distance, but only the sign (Mao, 2016). The main idea of LSGAN is to use the least squares loss function which provides a smooth and non-saturating gradient in the discriminator. This loss function penalises datapoints that are quite far away from the decision boundary, proportionally to the distance; even though they have been classified correctly. We can represent the loss as seen in *Equation 2* and *Equation 3* below:

$$\begin{gathered}\min_D V_{LSGAN}(D) \qquad (2)\\ = \frac{1}{2}\mathbb{E}_{x\sim p_{data}(x)}[(D(\mathbf{x})-b)^2] + \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[(D(G(\mathbf{z}))-a)^2],\end{gathered}$$

$$\min_G V_{LSGAN}(G) = \frac{1}{2}\mathbb{E}_{z\sim p_z(z)}[(D(G(\mathbf{z}))-c)^2] \qquad (3)$$

Here, the $a-b$ coding scheme is used for the discriminator (a for fake and b for real data), the most optimal value for $a-b$ being 0 and 1 respectively. Moreover, the parameter c denotes the proportion of G generated images that are meant to be classified as real by D. The value for this parameter was chosen to be 1 since we want to mislead the discriminator to the fullest extent possible.

Two main advantages of using LSGAN over standard GAN is that the latter is able to generate higher quality images and perform more stable during the learning process (Mao, 2016). These were our main reasons to follow this approach in our baseline model. In our future research, we would like further our understanding and explore other alternatives to make the training process of GANs more efficient, such as Wasserstein GAN (WGAN)(Gulrajani, 2017).

Moreover, we used the Adam optimiser which has been already introduced in the previous semester (Kingma & Ba, 2014). This was motivated by recent studies (Radford, 2015) in which Adam helps to accelerate training in most GANs. All the parameter values for Adam were set as default (i.e. $\beta_1 = 0.9, \beta_2 = 0.999$, $\epsilon = 10^{-8}$).

4.2. Adaptive Dropout

Dropout allows a single deep neural network (DNN) to behave similarly to an average of multiple networks, without the time constraints involved in training multiple DNNs. This is achieved by randomly dropping hidden units and their connections during training and processing the mini-batch through the altered network. This prevents individual nodes from co-adapting to other nodes in the network, as they cannot rely on these nodes always being present, making the hidden units more robust to missing features. Dropout then samples from an exponential amount of smaller networks. The predictions made by these networks are averaged and used at test time. Dropout has been seen to improve performance in supervised learning tasks such as vision, speech recognition, and image classification (Srivastava et al., 2014).

However, dropout uses a constant hyper-parameter for the probability of discarding a unit. This model does not take into account any characteristics of individual hidden units when deciding whether or not they should be omitted. (Srivastava et al., 2014). Ba and Frey state that this is of particular importance when considering hidden units which are capable of individually predicting the presence or absence of particular features (Ba & Frey, 2013). They further conclude that dropout does not deliver any performance improvements in the case of unsupervised feature learning with a network trained from scratch. Ba and Frey achieve a 13 percent error rate when training a fully connected DNN with two hidden layers on the NYU Object Recognition Benchmark (NORB) dataset. They obtain identical results with the addition of dropout (Ba & Frey, 2013).

The authors propose an alternative regularisation technique referenced as standout or adaptive dropout. A binary belief network is overlayed on the DNN, and trained concurrently, in order to stochastically adapt its architecture depending on the input (Ba & Frey, 2013). The output variable distribution over the hidden mask variables can be vied as a Bayesian posterior distribution, rather than a marginalisation of the hidden mask variables over a latent model, as is often the case in a traditional belief network (Ba & Frey, 2013). It is worth noting that posterior distributions generated using traditional Bayesian inference are independent of the test input, contrary to the posterior distribution in our adaptive dropout network. Ba and Frey argue the viability of this model based on the strong correlation which exists between the connectivity and weight magnitudes of adjacent layers in a neural network.

The activity of a unit j in a DNN is denoted by a_j, with its inputs referenced as $a_i : i < j$. Therefore, in dropout, a_j is set to 0, with probability equal to the predefined exclusion probability hyper-parameter. Adaptive dropout instead introduces a binary variable m_j, which is used to mask a_j. Therefore a_j can be defined as follows:

$$a_0 = 1, \qquad (4)$$

$$a_j = m_j g(\sum_{i:i<j} w_{j,i}\, a_i), \qquad (5)$$

where $w_{j,i}$ references the weights from unit i to j and $g(x)$ refers to the activation function used (Ba & Frey, 2013). Next, we introduce the logistic function $f(x)$:

$$f(x) = \frac{1}{1+e^{-x}}, \qquad (6)$$

which is used to define the adaptive dropout probability, dependent on the input activities. This is defined as follows:

$$P(m_j = 1|\{a_i : i < j\}) = f(\sum_{i:i<j} \pi_{j,i}\, a_i), \qquad (7)$$

where $\pi_{j,i}$ refers to the weights from unit i to j in the adaptive dropout network. Finally, similarly to standard dropout, we compute the expectation of *Equation 5* in order to process an input using the stochastic feed-forward process as follows:

$$\mathbb{E}[a_j] = f(\sum_{i:i<j} \pi_{j,i}\, a_i) g(\sum_{i:i<j} w_{j,i}\, a_i) \qquad (8)$$

Ba and Frey perform a series of further experiments which involve training auto-encoders on the NORB data set and using a linear classifier to predict the object class from the extracted features. They achieve a 18 percent relative reduction in error rates when using adaptive dropout, when compared to its fixed counterpart and are able to successfully train deep auto-encoders from scratch, without the need for layer-by-layer pre-training.

Data Set	loss_da	loss_db	loss_ga	loss_gb
T & C Easy	0.1757	0.1776	0.2395	0.2596
T & C Hard	0.1729	0.1774	0.3523	0.2668
A & K Easy	0.1744	0.1776	0.2454	0.2396
A & K Hard	0.1717	0.1710	0.3603	0.3773

Table 1. Averages loss in baseline experiments 1,2,3,4 - T & C refers to Trump (Face A) & Clinton (Face B), A & K refers to Albert (Face A) and Karol (Face B)

We will implement the adaptive dropout binary belief network in our auto-encoders in order to examine the impact of this on the performance of our model.

5. Baseline Experiments

Hyperparameters: All experiments use a learning rate of 0.0001 for the discriminator and generator. Experiments 1, 2 and 3 use a mini-batch size of 128 and experiment 4 uses a mini-batch size of 32 due to the small size of the dataset. All experiments are run on 5000 iterations, due to time constraints.

The average losses for the discriminator (ADL) and generator (AGL) are summarised in *Table 1.*

Experiment 1 - Trump Clinton Easy

Aim: This experiment was carried out using the default model described in *Section 4.1* and the dataset described in *Section 3.2* . The experiment aims to understand the behaviour of the model when trained with a less challenging dataset.

Results: The average generator loss is similar for both faces, 0.2395 for Trump and 0.2596 for Clinton. Although the loss function for the generator is unstable between consequent iterations, the general trend is that the loss decreases throughout training. The image translation generated by this model (*Figure 2a* & *Figure 2b*) is not ideal, however, the features of Donald Trump can be clearly seen on the final image. The average discriminator loss does not change throughout training and remains constant at 0.1757 for Trump and 0.1776 for Clinton.

Conclusions: The decreasing trend of the generator loss functions indicates that training is taking place effectively. Furthermore, the overlapping of the loss functions for the Trump and Clinton image sets show that they both provide a similar level of training difficulty. This result is logical when the nature of the data set is considered (*Section 3.2)*, as both sets of images are sourced from the same video. Furthermore, the instability of the generator loss functions can be attributed to the nature of the model, as similar patterns are observed in other experiments (Allen & Li, 2016). The relatively low quality of the final image can be attributed to the short training time of the model, as the code-base recommends training for 50000 iterations, 10 times more than our experiment.

Experiment 2 - Trump Clinton Hard

Aim: This experiment was carried out using the default model described in *Section 4.1* and the dataset described in *Section 3.1.* The aim of this experiment is to investigate the impact of inhibiting factors such as varying backgrounds, lighting conditions and watermarks on the training process. We also aim to assess the impact of the smaller diversity of the Clinton image set on the training process. These results will be compared with those achieved in Experiment 1.

Results: The average generator loss differs significantly for both faces, 0.3523 for Trump and 0.2668 for Clinton. The generator loss function follows the same decreasing trend as in Experiment 1 and is similarly unstable between consequent iterations. The image translation generated by this model (*Figure 2c* & *Figure 2d*) is inferior to that of Experiment 1, however, some features of Trump such as the eyes and lips can be observed on the transformed image. The average discriminator loss does not change throughout training and remains constant at 0.1717 for Trump and 0.1710 for Clinton, marginally lower than in Experiment 1.

Conclusions: The significantly lower generator loss for Clinton can be attributed to the smaller pool of sources used to create the image set and the presence of 255 images taken from a single source. The 0.2668 loss can be compared to the losses seen previously, indicating that the model found training on the Clinton image-set as difficult as on the sets in Experiment 1. However, training on the Trump image-set proved to be significantly harder, which can explain the decline in quality of the final translation.

Experiment 3 - Karol Albert Easy

Aim: This experiment was carried out using the default model described in *Section 4.1* and the dataset described in *Section 3.5* . The experiment aims to confirm the results obtained in Experiment 1 using a different dataset with similar features (image characteristics, number of images).

Results: The average generator loss (AGL) is similar for both faces, 0.2454 for Albert and 0.2396 for Karol. As seen in the previous two experiments, the AGL decreases throughout training. The quality of the image translation generated by this model (*Figure 3a* & *Figure 3b*) significantly surpasses that of previous experiments and looks relatively realistic. The average discriminator loss does not change throughout training and remains constant at 0.1744 for Albert and 0.1776 for Karol, which is very similar to the values seen in the previous experiments.

Conclusions: The overlapping of the loss functions for the Albert and Karol image-sets show that they both provide a similar level of training difficulty, which is consistent with the pattern seen in Experiment 1. Therefore, this can be seen as confirmation that the lack of factors which inhibit training leads to lower AGL and more consistent results. The superior quality of the image, when compared to Experiment 1., can be related to the similar facial features and same-sex of the two subjects. Furthermore, this can also be linked to the data set from Experiment 1. being extracted from an external source and reshaped to the 300 x 300 input size, which resulted in the slight deformation of the images.

Experiment 4 - Karol Albert Hard

Aim: This experiment was carried out using the default model described in *Section 4.1* and the dataset described in *Section 3.4* . The experiment aims to investigate the behaviour of the model when trained on a significantly harder dataset (further details in *Section 3.4*) than in Experiment 2, which also contains notably fewer images.

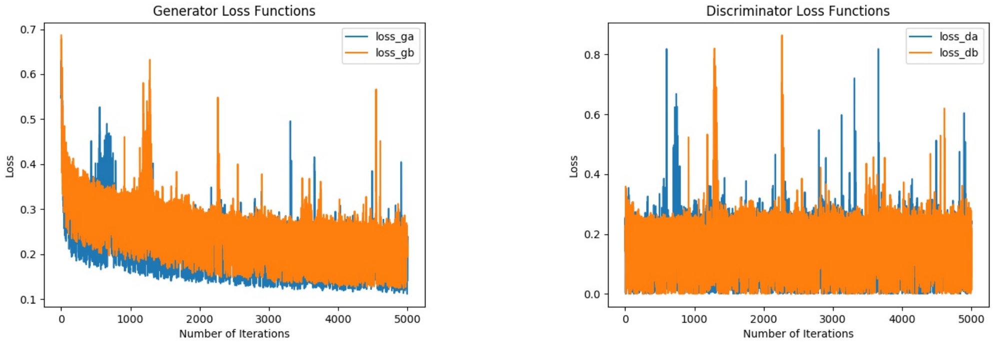
Results: The average generator loss (AGL) is similar for both faces, 0.3603 for Albert and 0.3773 for Karol. As seen in the previous three experiments, the AGL decreases throughout training. The quality of the image translation generated by this model *Figure 3c* & *Figure 3d* is the worst seen so far. The average discriminator loss does not change throughout training and remains constant at 0.1717 for Albert and 0.1710 for Karol, which is very similar to the values seen in the previous experiments.

Conclusions: The overlapping of the loss functions for the Albert and Karol image sets show that they both provide a similar level of training difficulty, which is significantly higher than in Experiment 1, and can only be compared with that of Trump in Experiment 2. This confirms the high training difficulty of this data set, which is due to the small number of images and high presence of factors which affect training, such as Snapchat and Facebook filters, varying backgrounds and angles as well as age differences. The lower quality of the final image translation can be explained by the aforementioned characteristics.

6. Interim Conclusions and Further Experiments

The following interim conclusions have been made:

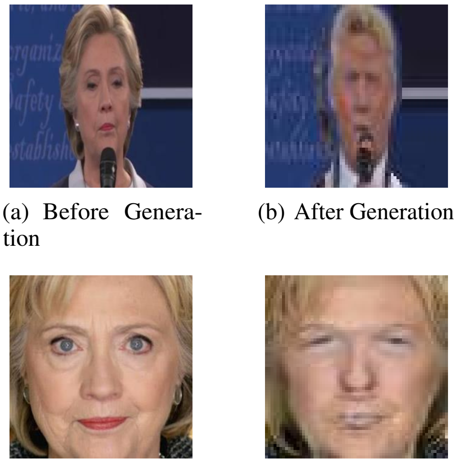
1. The model achieves satisfactory performance when trained on *easy* data over 5000 iterations. However, the resulting images still leave room for improvement when compared to results obtained in research (Dong et al.,2017). We believe this would require



(a) Generator Loss on Data Set B (Trump and Clinton Easy)

(b) Discriminator Loss on Data Set B (Trump and Clinton Easy)

Figure 1. Loss funtion for Experiment 1 (*loss_ga* and *loss_da* indicate generator and discriminator losses respectively for the Trump Sample, *loss_gb* and *loss db* indicate losses for the Clinton Sample)



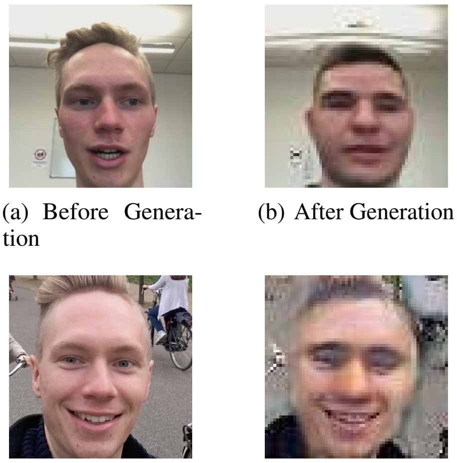
(a) Before Generation

(b) After Generation

(c) Before Generation

(d) After Generation

Figure 2. Experiment 1 and Experiment 2 Image Transformation



(a) Before Generation

(b) After Generation

(c) Before Generation

(d) After Generation

Figure 3. Experiment 3 and Experiment 4 Image Transformation

longer training runs, as suggested by the original code-base (50000 iterations) (GitHub - shaoanlu, 2018). We will perform these experiments in our further research.

2. The generator loss functions exhibit a declining trend in all experiments, indicating the progress in training. All discriminator average losses remain consistent across training. We cannot provide a justification for this behaviour and will aim to deepen our understanding of this and provide further details in the final report, however, similar patterns can be seen in experiments conducted by Allen and Li (Allen & Li, 2016).

3. A pool of images originating from the same source in a data set results in lower training difficulty, as seen in the case of Clinton in Experiment 2.

4. Factors such as varying backgrounds, lighting conditions, angles, the presence of watermarks and image distortion filters all result in a higher training difficulty and subsequently lower quality of the final translation.

5. Varying gender of the subjects has a negative impact on the quality of the final image transformation, as seen when comparing the results of Experiments 1. and 3. Dong manages to achieve satisfactory results in swapping the faces of different genders, therefore our model still has room for improvement (Dong et al.,2017). Furthermore, image deformation of the dataset used in Experiment 1. resulted in poorer results.

6. A reliable metric must be provided in order to assess the quality of the final swapped images. This also makes it difficult to compare our results with the literature. We will aim to train a linear classifier, which behaves similarly to the discriminator described in *Section 4.1.3* and classifies images as either fake or real.

7. Further investigation is required in order to understand the instability of the loss functions. We will aim to deepen our understanding of this. Furthermore, if time permits, we will attempt to experiment with other loss functions, such as WGAN (Gulrajani, 2017).

8. The current input size of 64 x 64 does not produce images of satisfactory quality and must be altered. We will aim to do this in our further experiments, nevertheless, higher input sizes might result in significantly extended training times.

9. Finally, we will rerun all of our experiments with the altered model, described in *Section 2.1*, in order to understand the impact of adding adaptive dropout, batch normalisation and various other improvements mentioned in *Section 2.1* on the performance of the model.

7. Potential Risks

The training process with 50000 iterations and a larger input image size is likely to take infeasible long to complete. Therefore, we might be forced to decrease one of these parameters or utilise AWS for some of our experiments. The difficulty of implementing adaptive dropout in our model has not been thoroughly investigated and might be potentially infeasible given the time constraints. Furthermore, we have not yet attempted to generate a video from our images, nevertheless, this can be done through the combination of still frames, and should not be a significant problem.

8. References

Ba, J. and P. Kingma, D. (2015). ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. ICLR.

Hinton, G., Srivastava, N. and Swersky, K. (2012). Neural Networks for Machine Learning - Lecture 6a Overview of Minibatch Gradient Descent.

Allen, A. and Li, W. (2016). Generative Adversarial Denoising Autoencoder for Face Completion.

Ba, J. and Frey, B. (2013). Adaptive dropout for training deep neural networks. [online] Available at: https://papers.nips.cc/paper/5032-adaptive-dropout-for-training-deep-neural-networks.pdf [Accessed 12 Feb. 2018].

Bengio, Y. (2016). Plug & Play Generative Networks: Conditional Iterative Generation of Images in Latent Space.

Bitouk, D., Kumar, N., Dhillon, S., Belhumeur, P. and Nayar, S. (2008). Face Swapping: Automatically Replacing Faces in Photographs. Columbia University.

Chintala, S., Denton, E., Arjovsky, M. and Mathieu, M. (2016). soumith/ganhacks. [online] GitHub. Available at: https://github.com/soumith/ganhacks [Accessed 16 Feb. 2018].

Chintala, S., Radford, A. and Metz, L. (2016). UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS. [online] Available at: https://arxiv.org/pdf/1511.06434.pdf [Accessed 12 Feb. 2018].

Dale, K., Sunkavalli, K., Johnson, M., Vlasic, D., Matusik, W. and Pfister, H. (2011). Video face replacement. ACM Transactions on Graphics, 30(6), p.1.

Doersch, C. (2016). Tutorial on Variational Autoencoders.

Dollar, P., Wojek, C., Schiele, B. and Perona, P. (2012). Pedestrian Detection: An Evaluation of the State of the Art. IEEE Transactions on Pattern Analysis and Machine Intelligence, 34(4), pp.743-761.

Dong, H., Neekhara, P., Wu, C. and Guo, Y. (2017). Unsupervised Image-to-Image Translation with Generative Adversarial Networks. [online] Available at: https://arxiv.org/pdf/1701.02676.pdf [Accessed 12 Feb. 2018].

GitHub - shaoanlu. (2018). shaoanlu/faceswap-GAN. [online] Available at: https://github.com/shaoanlu/faceswap-GAN [Accessed 12 Feb. 2018].

Goodfellow, I. (2014). Generative Adversarial Networks.

Goodfellow, I. (2016). Improved Techniques for Training GANs.

Gulrajani, I. (2017). Improved Training of Wasserstein GANs.

Hern, A. (2018). Reddit bans 'deepfakes' face-swap porn community. [online] the Guardian. Available at: https://www.theguardian.com/technology/2018/feb/08/reddit-bans-deepfakes-face-swap-porn-community [Accessed 16 Feb. 2018].

Hinton, G. (2009). Deep belief networks. Scholarpedia. [online] Available at: http://www.scholarpedia.org/article/Deep_belief_networks [Accessed 12 Feb. 2018].

Hoffman, J., Tzeng, E., Park, T., Zhu, J., Isola, P., Saenko, K., Efros, A. and Darrell, T. (2018). CyCADA: Cycle-Consistent Adversarial Domain Adaptation. ICLR 2018.

Jansenh.stackstorage.com. (2017). STACK. [online] Available at: https://jansenh.stackstorage.com/s/UayUugaE0GSda0y?dir=

Karpathy, A. (2015). Convolutional Neural Networks for Visual Recognition.

Karpathy, A., Abbeel, P., Brockman, G., Chen, P., Cheung, V., Duan, R., Goodfellow, I., Kingma, D., Ho, J., Houthooft, R., Salimans, T., Schulman, J., Sutskever, I. and Zaremba, W. (2016). Generative Models. [online] OpenAI Blog. Available at: https://blog.openai.com/generative-models/ [Accessed 15 Feb. 2018].

Korshunova, I., Shi, W., Dambre, J. and Theis, L. (2017). Fast Face-swap Using Convolutional Neural Networks. [online] Available at: https://arxiv.org/pdf/1611.09577.pdf [Accessed 12 Feb. 2018].

Ledig, C. (2017). Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network.

Mao, X. (2016). Least Squares Generative Adversarial Networks.

Metz, L. (2016). Unrolled Generative Adversarial Networks.

NBC News. (2016). The Second Presidential Debate: Hillary Clinton And Donald Trump (Full Debate) | NBC News. [online] Available at: https://www.youtube.com/watch?v=FRlI2SQ0Ueg&t=272s [Accessed 12 Feb. 2018].

Ng, A. and Jordan, M. (2014). On Discriminative vs Generative classifiers : A comparison of logistic regression and naive Bayes.

Radford, A. (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks.

Shi, W., Caballero, J., HuszÃąr, F., Totz, J., Aitken, A., Bishop, R., Rueckert, D. and Wang, Z. (2018). Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network.

Suwajanakorn, S., M. Seitz, S. and Kemelmacher-Shlizerman, I. (2015). What makes Tom Hanks look like Tom Hanks. 2015 IEEE International Conference on Computer Vision.

Thewlis, J., Bilen, H. and Vedaldi, A. (2017). Unsupervised learning of object frames by dense equivariant image labelling. [online] Available at: https://arxiv.org/pdf/1706.02932v2.pdf [Accessed 12 Feb. 2018].

Zhu, J. (2017). Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks.