Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology

Sondre Langesæter Solberg

Influence of Image Effects and Filters in Deepfake Detection

Master's thesis in Information Security
Supervisor: Lasse Øverlier
June 2022

NTNU
Norwegian University of
Science and Technology

Sondre Langesæter Solberg

Influence of Image Effects and Filters in Deepfake Detection

Master's thesis in Information Security
Supervisor: Lasse Øverlier
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology

NTNU
Norwegian University of
Science and Technology

Abstract

Videos have become an important part of our digital society as a source for information and entertainment. In accordance with this digital development, new ways of editing and manipulating videos have emerged. Deepfakes are an example of this, where images and videos of people are manipulated and edited to appear as something they are not. This can have major consequences if used for malicious use. Good systems for detecting deepfakes are thus important.

Deepfake detection is a field that has received a lot of focus in recent years, and a lot of resources have been invested to accelerate development. This has resulted in large deepfake datasets and various detection models. These models are promising, but there is little research on how visual influences such as various image filters affect the models.

This thesis studies how state of the art deepfake detection models generalize to new data with various filters and effects. To figure this out different experiments have been performed on the models with custom data applied various filters and effects. The data gathered from the experiments gives insight into which filter or effect are the most and least influential for the detection results.

Sammendrag

Videoer har blitt en viktig del av vårt digitale samfunn, som en kilde for informasjon og underholdning. I samsvar med denne digitale utviklingen har det kommet nye måter å redigere og manipulere videoer. Deepfakes er et eksempel på dette, hvor bilder og videoer av mennesker blir manipulert or redigert til å fremstå som noe de ikke er. Dette kan ha store konsekvenser hvis det brukes til ondsinnet bruk. Det er derfor viktig å ha gode systemer for å oppdage deepfakes.

Deepfake deteksjon er et område det har vært mye fokus på de siste årene, og det er brukt mye ressurser for å få fart på utviklingen. Dette har resultert i store deepfake datasett og forskjellige deteksjonsmodeller. Disse modellene er lovende, men det er lite forskning på hvordan forskjellige bildefiltre og effekter påvirker resultatene til modellene.

Denne oppgaven studerer hvordan moderne deepfake deteksjonsmodeller generaliserer til ny data med ulike filtre og effekter. For å få et svar på dette er det blitt utført forskjellige eksperimenter på modellene med tilpassete datasett bestående av forskjellige filtre og effekter. Resultatene og dataen fra disse eksperimentene vil gi innsikt i hvilke bildefiltre og effekter som påvirker resultatet både minst og mest.

Acknowledgments

First of all i want to thank my supervisor Lasse Øverlier for helping me throughout the project by providing good ideas and feedback on the thesis. I also want to thank Lars Erik Pedersen for providing access to GPU resources at NTNU SkyHigh, making it possible to perform my experiments.

I addition a special thank you to my mother and father for supporting me throughout the thesis and my studies. Also a thank you to my girlfriend for support and motivation.

Contents

Figures

Tables

Abbreviations

- DFDC - Deepfake Detection Challenge
- AUC-ROC - Area Under the Receiver Operating Characteristic Curve
- WS-DAN - Weakly Supervised Data Augmentation Network

Chapter 1

Introduction

We live in a society that is dependent on digital media consumption. The mobile phone is always accessible and the flow of information from social media and the news is endless. Popular apps like TikTok, Snapchat and Facebook are all used to share information in one way or another. Anyone can publish information, and credibility often becomes an afterthought. This is problematic, as false and misleading information can be shared carelessly. The publishing of manipulated media such as deepfakes is a cause for concern, as it can be used for malicious purposes suitable of doing great harm. It is clear that deepfakes can be a big problem for the society, and its important to have tools in place that can detect them.

1.1 Topics Covered by the Project

Deepfake is an editing and manipulation method uses to change a persons face, either to a whole new identity or to make the person appear as someone they are not. The state of the art deepfakes are becoming very realistic, and if used for the wrong purposes can be very harmful. Detecting deepfakes is thus important and developing systems to detect them is a difficult task. A lot of research have been done and are ongoing in the field of machine learning and deepfake detection.

One of the goals of this project is to explore the state of the art deepfake detection methods that exist, test them out and experiment with new deepfake data containing common image filters and effects. This will provide an answer on how the deepfake detection models generalize to different kinds of data, in addition to providing an insight into what kind of effect has the most and least impact on the detection rate.

1.2 Keywords

Deepfake Videos, Deepfake Detection, Image Filters, Image Effects, Deepfake Dataset

1.3 Problem Description

Much of today´s information is consumed through videos, this can be anything from short clips on the mobile phone to longer reports on the TV. People actually rely more on the information provided through videos than many others types of sources [1]. With the advancement of deepfake videos, a technology that is accessible to the general public, there is a high risk of malicious and harmful use. The state of the art deepfakes have also become so good that it is often difficult to interpret whether a video is fake or not. Without statistics to support it, it is likely that much of today´s society is not aware of what deepfakes are, and how it can affect them. Also, social media platforms does not seem to have any systems in place to detect or flag if deepfake videos are uploaded. This probably makes it easy to fool ordinary people who have no knowledge of what deepfakes are. People who are familiar with deepfakes are probably more critical, but because the quality of deepfakes have become so good, they can most likely be fooled as well. The importance of being able to detect deepfakes both reliably and accurately is clear. It could be discussed that manipulated media should be watermarked when published online.

1.4 Justification, Motivation, and Benefits

Editing and manipulating videos is far from new and is something that has been going on for a long time, especially in the film industry. Deepfakes, on the other hand are a modern way of manipulating videos using machine learning, where the identify of one person is exchanged with another. This can create completely new identities or falsify existing identities. Deepfakes are often used to make funny and innocent videos, where two famous persons identities are replaced with another, and there is a shared understanding that this is fake. For example, there is a TikTok user dedicated to creating deepfake parodies of Tom Cruise, called DeepTomCruise [2]. The problem however, is when deepfakes are used to falsify actions and statements, something that can be very harmful.

There are countless examples of humiliating and harmful deepfakes. For example, there was used a voice deepfake to trick a CEO into transferring funds to a scammer [3]. Additionally, it has become a major problem with adult videos. Women´s faces and identities are used in such videos without consent [4]. In conjunction with the war in Ukraine, a Deepfake was published of President Zelenskyy in which the Ukrainian people are told to surrender [5]. The deepfake was broadcasted on a hacked Ukrainian TV channel, and was quickly distributed online. Fortunately, the deepfake was exposed relatively quick as it was not perfect, but it shows the severity and one can only imagine what would have happened if it had fully succeeded in its purpose. Given how harmful deepfakes can be, it is clear how important it is to have systems in place that can detect them.

1.5 Research Questions

The purpose of this project is to perform different experiments in order to find out how deepfake detection models are affected by different image filters and effects. In conjunction with this, the following research questions have been formulated:

1. How does existing deepfake detection models generalize to new data with common image filters and effects?
2. Which image filter or effects are the most influential in the detection of deepfakes.
3. Which image filter of effects are the least influential in the detection of deepfakes?

1.6 Planned Contributions

Currently, the state of the art deepfake detection methods are not accurately or good enough to be used in automatic systems, and proper deepfake detection is considered an unsolved problem [6]. Every day, countless amounts of videos are published on the internet, how many of these are deepfakes are impossible to say, but the vast majority are not deepfakes. Videos published on the internet and on social media are affected by compression and results in videos with lower quality [7]. Social media also provides the ability to edit photos and videos that are uploaded with various filters and effects. The purpose of this project is to contribute with data and information on how state of the art deepfake detection models generalize to common image filters and effects. Custom datasets applied various image filters have been created, and different experiments have been performed on the data. There are the results from the experiments that will provide information on how the different image filters and effects influence the detection rate of deepfakes.

1.7 Ethical and Legal Considerations

In order to perform experiments in this project, a lot of deepfake data is needed. The problem however is that a large number of deepfakes are made without consent and approved use. It is therefore important that data used in this project is legal to use, and have agreeing participants. This project uses data from the Deepfake Detection Challenge Dataset [8] as well as data from DeeperForensics-1.0 [9]. both of these datasets meet the requirements of agreeing participants and legally authorized use.

1.8 Thesis Structure

This thesis consist of 7 chapters. It starts with the first chapter **Introduction** which explains the purpose of the project with corresponding research questions. Followed by **Background** which provides background information for the topic, and study what research and information that are already present. In chapter three the **Methodology** is presented, which explains the experimental design and how the research questions will be answered. Continued by the **Experiments** chapter which explains how the experiments where performed and why various choices were made. Then in chapter five, the **Results and Analysis** from the experiments are presented. With **Discussion** of different thoughts and findings from the experiment results in chapter sixth. Lastly in chapter seven **Conclusion**, it is concluded what the findings have achieved in accordance with the research questions.

Chapter 2

Background

The origins of deepfakes occurred in November 2017 when manipulated videos were uploaded to a Reddit forum by a user named Deepfakes[10]. The term deepfake is a pun from the machine learning concept Deep Learning and fake videos [11]. They are made using machine learning techniques, and there has been great progress in recent years, which has made Deepfakes much better and more realistic. After the appearance of deepfakes, there has been more and more focus on how they can properly be detected. Fortunately, a large number of deepfakes are of such poor quality that it is easy to see that they are fake. The problem however, is good deepfakes as these are much more difficult to detect with human judgment. It is thus important to have systems that can detect deepfakes of both good and poor quality.

This section aims to give a simple overview of how deepfakes are created to provide a basic understanding of the technology. In addition, related work within deepfake detection will be studied to give a basis for the experiments and goals of this project.

2.1 DeepFake Creation

There are various tools available for creation deepfakes, whereas the complexity and quality they offer varies a lot. Many of them are designed to be simple to use as an entertainment tool for creating funny videos, without trying to fool anyone, and it is clear that the video is fake. The more advanced tools, on the other hand, often allow for greater functionality and complexity, resulting in deepfakes that are much more realistic.

Today, most deepfakes are made using Generative Adverserial Networks - GAN, a deep learning technique from the world of machine learning[12]. A thorough review of GAN´s is out of scope of this thesis, but simply explained; The method involves training two models, a generative and a discriminative model, whereas the goal of the generative model is to trick the discriminative model into thinking that the data it outputs is part of the training data[12]. In the case of deepfake videos, the discriminative model is trained to recognize deepfakes and accept real faces. The generative model creates and outputs deepfakes with the goal of tricking the discriminative model into accepting them. For every iteration, the generative model improves the deepfake until it is accepted by the discriminative model. Figure 2.1 shows a simple illustration of the basic concept of a GAN framework.

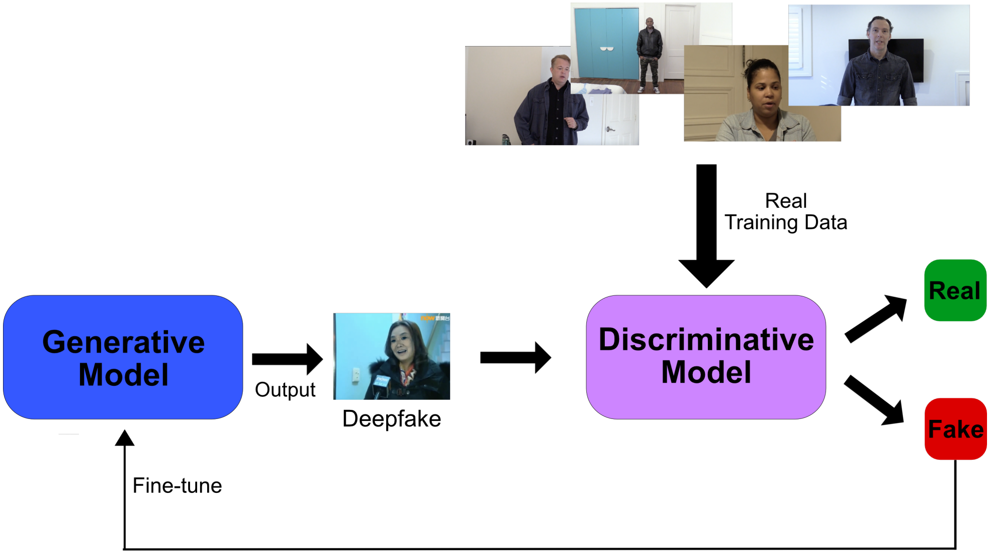


Figure 2.1: Illustration of a simple Deepfake Generative Adverserial Network (GAN) pipeline. (Example "Deepfake" from DeeperForensics dataset [9], and example of "Real Training Data" from DFDC dataset [8].)

The website "thispersondoesnotexist.com" uses StyleGAN2 to create high quality deepfake photos of people who do not exist [13]. The deepfake photos are so realistic and detailed that it is hard to believe they are fake.
There exist many applications for making deepfakes of which some of the most popular are the following;

- DeepFaceLab [14]
- FSGAN - Face Swapping GAN [15]
- StyleGAN3 [16][17]
- Faceswap [18]
- Deepfakes Web [19]
- Reface - Android/iOS [20]
- ZAO App Deepfake - Android/iOS [21]

Major social media companies such as Snapchat and TikTok have also implemented deepfake technology as an creative tool in their applications [22]. In addition, there exist a numerous amount of low effort deepfake apps for smartphones, as well as lip sync apps that mimics mouth movement to text and songs, such as WOMBO [23].

2.2 Deepfake Dataset

The development of deepfake detection models is only possible with access to proper and large deepfake datasets. Hence various datasets have been designed for this purpose. This section will study two different datasets that are appropriate for this project.

2.2.1 DFDC - Deepfake Detection Challenge Dataset

The Deepfake Detection Challenge Dataset was created by Facebook with the goal of accelerating the research on deepfake detection [8]. The dataset was created in accordance with a deepfake detection competition on Kaggle [24]. There exist several other deepfake datasets, but according to Facebook these are deficient, either in the amount of data, the amount of subjects, or in the form of deepfake methods used [8]. It was also important for the creators that all participants approved the use of the dataset, and agreed on their face and identity being manipulated.

The data from DFDC is publicly available and can be downloaded from either Kaggle [25], or Facebook AI [26]. It must be noted that only the training data is available on Kaggle, test and validation data has been published after the competition and is only available on Facebook AI [26].

The dataset consist of both real videos and deepfakes, whereas the deepfakes have been created with various deepfake methods. This was done to cover a wide range of popular deepfakes methods and to give a more varied dataset [8]. The different deepfake methods used in the DFDC dataset are [8]:

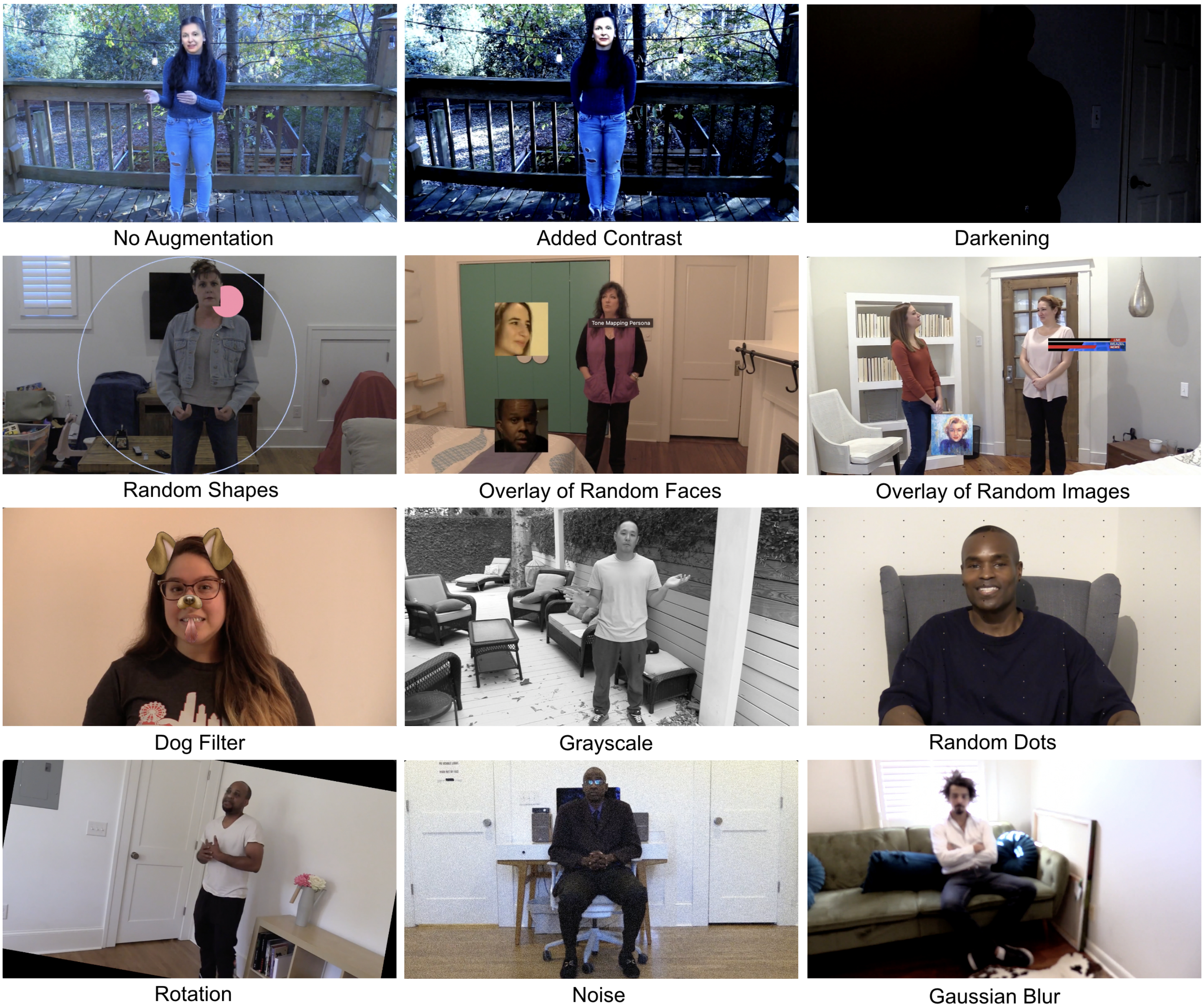
- Deepfake Autoencoder (DFAE)
- MM/NN Face Swap
- NTH
- FSGAN
- StyleGAN

Some of the videos in the dataset are also applied various augmentations. Augmentation is a machine learning method used to create a larger dataset by making changes to the data that is already present. This is done to learn the machine learning models to adapt to variations of the data that it is likely to encounter, as a way to generalize better [27].

In the DFDC dataset, a total of 18 different augmentations have been applied [8]. The augmentations used are:

- Gaussian Blur
- Noise
- Grayscale
- Horizontal Flipping
- Rotation
- Resolution Changes
- Contrast Changes
- Brightening/Darkening
- Frame-rate Changes
- Audio Removal
- Encoding Quality Changes
- Flower Filter
- Dog Filter
- Random Faces Overlay
- Random Dots
- Random Image Overlays
- Random Shapes Overlays
- Random Text Overlays.

Figure 2.2 shows different examples of augmentations used in DFDC. In the dataset, the augmentations are applied at different levels, hence why the darkening example is considered extreme.



No Augmentation

Added Contrast

Darkening

Random Shapes

Overlay of Random Faces

Overlay of Random Images

Dog Filter

Grayscale

Random Dots

Rotation

Noise

Gaussian Blur

Figure 2.2: Example of augmentations from DFDC dataset [8]. Note: The darkening example is considered extreme.

In this project various custom datasets with different filter and effects have been created in order to perform different experiments. Some of these effects are similar to the augmentations used in the DFDC dataset, and these are [8]; Gaussian Blur, Noise, Grayscale, Horizontal flipping, Rotating, and Resolution changes.

The DFDC dataset consist of train, test, validation and a preview set. The preview set is a smaller version of the full DFDC dataset and consist of 5.000 videos, split between a test and a training set [28]. In the Deepfake Detection Challenge, the validation set was used to rank the models on the public leaderboard on Kaggle[8][24]. Whereas the test set was used to rank the models on the private leaderboard, which was used to select the competition winner [8].

The full dataset consist of [8]:

- Training set:
  - 119.154 videos
  - 83.9% of the videos are deepfakes
  - 16.1% of the videos are real videos
  - 4 different deepfake methods
  - 486 different identities
  - No augmentations
- Validation set:
  - 4.000 videos
  - 50% of the videos are deepfakes
  - 50% of the videos are real videos
  - 214 different identities
  - 5 different deepfake methods
  - 79% of all the videos where altered with augmentations
- Test set:
  - 5.000 videos
  - 50% of the videos are deepfakes
  - 50% of the videos are real videos
  - 260 different identities
  - 4 different deepfake methods
  - 79% of all the videos where altered with augmentations
  - Note: Originally, the test set consisted of 10.000 videos, but half of the videos where content taken from the internet, and was removed before Facebook published the dataset.

2.2.2 DeeperForensics Dataset

The DeeperForensics dataset is another contribution to the research on deepfake detection [9]. The creators point out that the data must represent real-world scenarios of deepfakes, with emphasize on quality, scale and diversity. Which means that the data must first and foremost be of a quality that represent the distribution of deepfakes in reality, have a large scale of videos to be able to train and test machine learning models, as well as have a diversity in the set that represent a variety of quality issues, such as noise, blur, compression, etc [9].

The total amount of videos in the dataset is 60.000, of which 10.000 are various deepfakes, and 1000 of these are original deepfakes without any augmentations [9]. The source videos used to make the deepfakes are collected from 100 different actors who have approved the use within the dataset [9]. The creators of DeeperForensics believe that good source videos are the key to great deepfakes and thus much of their focus have been on generating a large variety of poses and expression from the actors in order to get a good basis for high quality deepfakes. The deepfakes in the dataset have been created with a deepfake method called Deepfake Variational Auto-Encoder (DF-VAE) [9].

The target videos used to create the deepfakes in the DeeperForensics dataset are 1000 videos taken from the FaceForensics++ dataset [9][7]. These videos are originally taken from YouTube and the rights to use them are unknown. Due to these unknown rights of use, these target videos are not a direct part of the DeeperForensics dataset and must be downloaded directly from FaceForensics++ [29][30]. Given that one of the goals with this project was to use ethically correct data, it was decided to not use data from FaceForensics++. Nevertheless, the manipulated deepfake videos from DeeperForensics are as mentioned based on the target videos from FaceForensics++, but the original identities have been replaced with the source videos from the approved actors. It can be discussed whether this is ethically good enough to qualify as approved use, but the identities are at least hidden. Figure 2.3 shows an example of a source video from a participating actor and a deepfake created from the source video.

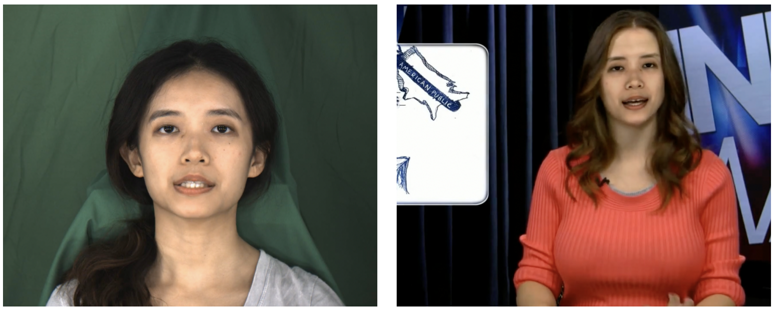


Figure 2.3: Example of source video (left) and corresponding deepfake (right) from the DeeperForensics dataset [9].

2.3 Deepfake Detection

As deepfakes have become more and more popular and accessible, the research and focus on deepfake detection is becoming more prominent. The better deepfakes gets, the harder and more important it becomes to detect them. In the recent years, a lot of resources have been invested in deepfake detection. Hence large datasets have been created to contribute in the space. The Deepfake Detection Challenge Dataset (DFDC) [8] and DeeperForensics Dataset are examples of this [9]. In conjunction with the creation of the DFDC dataset, a competition has been arranged to make advancements in the deepfake detection space.

2.3.1 Deepfake Detection Challenge

At the end of 2019, the Deepfake Detection Challenge was hosted on Kaggle, presented by Facebook with several partners [24]. The competition took place over 4 months with a total of 1 million dollars in the prize pool. More than 2.000 teams participated, creating over 35.000 machine learning models [31]. The goal of the competition was to use the associated DFDC Dataset and create the best deepfake detection model. The arrangement of the Deepfake Detection Challenge was a big leap forward for deepfake detection, as it created a lot of focus and engagement around the topic. The data from DFDC is publicly available, promoting further work and research, which this project is an example of.

Results - Deepfake Detection Challenge

The models were measured against a public leaderboard where the log loss score of the models was used for the ranking [8]. After the competition was over, the final results were calculated on a private leaderboard with a secret test-set the public weren't aware of [8]. The final result and the winning model ended at 65% accuracy on the private leaderboard[31].

2.3.2 Deepfake Detection Challenge - Models

The Deepfake Detection Challenge has created massive traction and research on deepfake detection models, and it thus makes sense to look at the top performing models. A detailed and thorough review of the models is out of the scope of this thesis.

1st Place Model - Deepfake Detection Challenge

The winning model on the Deepfake Detection Challenge was created by Selim Seferbekov, achieving the best results on the private leaderboard [8]. The model scored 82.56% accuracy on the public leaderboard, and 65.18% accuracy with a log loss 0f 0.42798 on the private leaderboard [31]. The log loss score for the public leaderboard was not available anymore.

The model uses a Multi-Task Convolutional Network (MTCNN) to detect faces [32], and EfficientNet to encoder various features from the faces [33] [34] [8]. The model is also trained on the training data from DFDC, with various custom augmentation applied at random levels [8][34].

Augmentation applied to training data [34]:

- Compression
- Gaussian Noise
- Gaussian Blur
- Mirroring / Horizontal Flipping
- Isotropic Resize
- Brightness and Contrast
- Black and White
- Rotation
- Removing Part of the face

2nd Place Model - Deepfake Detection Challenge

The second place model from the Deepfake Detection Challenge was created by the team WM consisting of three team members, Wenbo Zhou, Hao Cui, and Hanqing Zhao [8][35]. The model scored a log loss of 0.2868 on the public leaderboard and 0.42842 on the private leaderboard [35][8].

This model uses Xception and EfficientNet to extract features from the faces [36] [33], and two Weakly Supervised Data Augmentation Network - WS-DAN [37] models to train the model on various data augmentations [35][8].
This model is also trained on the training data from DFDC, and have in addition to WSDAN been applied various other augmentations as well [35].

Augmentation applied to training data [35]:

- Mirroring Horrizontal flipping
- Gaussian Noise
- Gaussian Blur and Motion Blur
- Hue-Saturation
- Brightness and Contrast changes
- Emboss and Image Sharpening
- Sepia Filter

Chapter 3

Methodology

Deepfake detection is an important topic where a lot of research has taken place in recent years. The challenges and difficulties in the field has not yet been properly solved, and there is still a lot of work and progress left to be done. The purpose of this project is to expand on previous knowledge and research, and answering the research question; 1. How existing machine learning models generalize to new deepfake data with different effects and filters. Research questions 2. and 3; Finding out which effects and filters are the most and least influential for the detection rate. The steps required to find an answer to these questions are to perform a series of experiments.

3.1 Experimental Design

The goal of the experiments are to gather data to analyze, and they can only be useful if they can provide accurate and valid results. It is therefore important to plan and conduct them in a proper way. To ensure this, an experimental design has been outlined. Consisting of **Experiment Hypothesis**, **Data Collection**, **Dataset Variations**, **Model Selection**, **Experiment Execution**, and lastly **Analyze Results**.

3.1.1 Experiment Hypothesis

Before proceeding with the experiments, it is important to have an idea of how the experiments will take shape and create an hypothesis for the expected results. A list of different filters and effects that would be interesting to test out where made, with a thought and hypothesis on how they will influence the detection rate. The filters were selected based on how common they are in videos, in addition to various rotations. Both models used in the experiments are made using Convolutional Neural Networks (CNN) [34][35], and the results from CNNs are in general prone to rotations the models have not been trained on [38]. It will therefore be interesting to see how the model adapts to various rotations.

No Filter

In order to have a basis to compare results against, it is important to perform an experiment on the original unmodified data that was collected. This data is expected to provide the best results, as it is has not been tampered with.

Black and White

Black and White filter is a very common filter that is widely used on the internet. It is therefore interesting to see whether this filter will have any effect on the detection rate, but probably to a very small extent, as it is only changing the color of the videos.

Gaussian Blur

The appliance of a blur effect is interesting as it is relatively common in videos of lower quality with poor focus and sharpness. The blur is expected to make the videos more unclear and will probably be a disturbing factor for the models. As the amount of blur can differ greatly, it has been concluded to perform three different experiments with varying degree of blur. Light, Medium and High blur, where all of them are expected to have a great impact on the results, whereas the highest level of blur will be the most influential.

Gaussian Noise

The use of added noise is also interesting as lower quality videos are often influenced by it. The added noise will most likely be a disturbing factor for the models, and have a significant effect on the results. It is still expected to perform better than the data with blurs, as the added noise is not as obscuring as a video with added blur. It would be interesting to test out different levels of noise, but because of time constraints it was not possible.

Mirrored

The mirroring of a videos is not expected to have any impact on the results, as it is such a simple effects that does not obscure or affect the original video in a meaningful way. It will be very surprising if the detection rate is affected.

Upside Down

Just as with the mirrored videos, turning them upside down does not alter the quality of the videos. It might have a impact on the results as the models are not trained on videos that are upside down, but it will be interesting to see how the models adapt.

90 Degree Rotation

Turning a video 90 degrees is also not altering the video quality, but it is possible to that the models may struggle a bit because the faces in the video are in a position they are not used to and haven´t been trained on. It is expected to have more impact on the results than the upside down videos.

Random Rotation

The videos will be rotated randomly from 5 to 45 degrees. To partially rotate the videos is not expected to have any meaningful impact on the results, as the models will probably generalize well to small rotation in the faces. From all the different rotation experiments, 90 degree is expected to be the most influential, followed by upside down, then random rotation, and lastly the mirrored videos will most likely be the least influential.

Lowered Resolution

Lowering of resolution is very common with videos uploaded to the internet, and it will therefore be interesting to see if this will have any affect on the results. The lower the resolution the more significant the impact on the results will most likely be, and the resolution will be lowered to a typical low quality video on the internet. This is expected to have a small impact on the results, as low quality videos have less details and more noise.

3.1.2 Data Collection

In order to perform the experiments and get credible results, there is a need for a proper deepfake dataset. The dataset must consists of both deepfake videos as well as regular non-manipulated videos. This means that the deepfake models must relate to both real and fake videos, which will give more accurate results that will not be affected by any bias the models may have. The data used can not be the same as the deepfake detection models have been trained on, something that would have resulted in over-fitting and a skewed results. Other considerations is that the data must be legally and ethically sound. This means that the participants in the videos from the dataset must have agreed on the use case.

3.1.3 Dataset Variations

When a proper dataset have been collected there is a need to create different variations of the dataset to perform the various experiments. For each filter and effect that will be tested, a unique version of the dataset is needed consisting of videos with the applied filter or effect.

3.1.4 Model Selection

The purpose of this project is to experiment with deepfake detection models in conjunction with various image filters and effects. It was therefore concluded that using already existing machine learning models would be the most beneficial for the experiments. Creating and training a custom machine learning model was not within the scope of the project, as the focus was on performing experiments. Additionally, the experiments will be run on two different models, as a way to have two reference points to base the results on. The models have primarily been selected based on availability, documentation, and performance.

3.1.5 Experiment Execution

In order to perform the experiments in a good and efficient way, a suitable environment and platform is needed. Both hardware and software compatible with the models are required, as well as enough storage space to accommodate the various datasets.

The experiments will be performed in parallel with the creation of the dataset variations. Meaning that when the first dataset is created it will immediately be run on the models. This is done to ensure that there are always results to refer to, in case time constraints would limit the amount of experiments done.

3.1.6 Analyze Results

When an experiment is performed, the model will output the classification results from the predicted deepfake data. The result data must be analyzed and studied in a convenient way. Meaning that suitable metrics are needed to properly evaluate the results. From these results it will be possible to compare how the different experiments performs in conjunction with each other, as well as evaluate and discuss why the results are as they are.

3.2 Limitations and Shortcomings

Performing experiments and creating dataset variations with different filters and effects is a time consuming process. As this project is time constrained, the amount of experiments possible to do is limited. It would be interesting to perform a lot more experiments with several more filters. Particularly, study how popular image filters from photo sharing apps like Instagram and TikTok would influence the results.

Chapter 4

Experiments

The main purpose of the project is to study deepfake detection in conjunction with different image filters and effects. This will be done by performing various experiments on deepfake detection machine learning models. The results will give an answer to the three research questions. How deepfake models generalize to new data with filters and effects, and which have the least and greatest influence on the detection rate.

4.1 Custom Dataset

There is a need to create a custom dataset in order to perform the experiments. The dataset that have been created is based on the test set from the Deepfake Detection Challenge Dataset (DFDC) [8], as well as the DeeperForensics-1.0 dataset [9]. One of the goals with this project was to carry it out in an legal and ethical manner, and it was thus important that the dataset used had participants who had approved the use of their identities in the dataset. Which is something both DFDC and the DeeperForensics datasets fulfilled [8][9].

It must be mentioned that only the deepfake videos from the DeeperForensics have been used in the custom dataset, as they follow this projects requirements with approval of use. Whereas the real videos from the DeeperForensics dataset are videos taken from the Internet, where the rights and approval of use are unknown [9]. For this reason they do not follow the ethical requirements for the project.

The reason why a combination of two datasets has been made is because a variation of different data will make the results more applicable to real world scenarios. Since the models used in this project are trained on the DFDC dataset, the inclusion of the DeeperForensics dataset was also done to even out any bias the models might have against the DFDC dataset. It is important to point out that the test data from DFDC is data the models have not been trained on. But many of the deepfake methods used to create the training and test set are the same, meaning that the models may perform better on the test set from DFDC than other data [8]. Something that the results from DFDC can confirm.

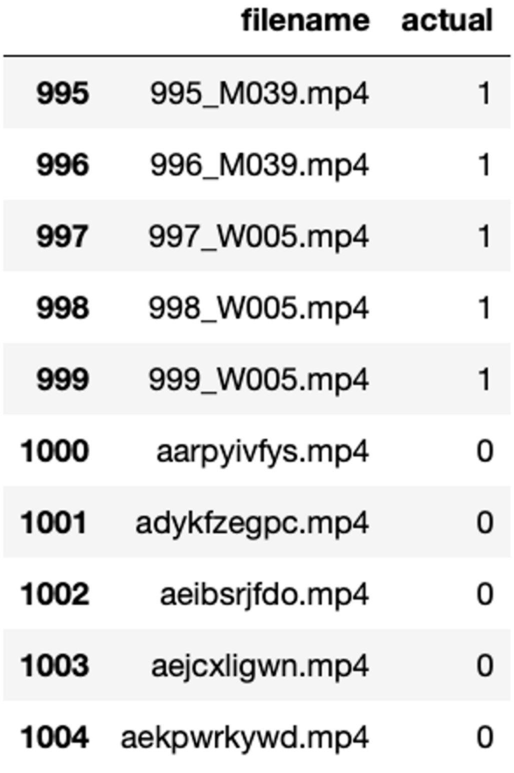
The models from DFDC where evaluated against two different datasets; a public test dataset which was the basis for the public leaderboard on Kaggle, as well as a private test dataset that was used to determine the winner [8][31]. The interesting part is that the best performing model scored 82.56% on the public test data, but the best performer on the private test data scored only 65.18% accurate[31]. An important detail that most likely is the reason for the large difference in results, is that 50% of the private test set consist of videos found on the Internet, while the other half was data created specifically for the Kaggle competition [8]. This is not the case for the public test set, which consist of videos made specifically for DFDC. This confirms the suspicion that the models performs best on similarly shaped data.

4.1.1 Dataset Structure

As one of the purposes of the project was to add custom filters and effects to the dataset and videos, it was important to have as original data as possible to work with, which means that all data with augmentations and effects were not relevant to use. From DeeperForensics, the choice fell on 1.000 manipulated deepfakes taken from the "reenact postprocess" part of the official dataset. This is deepfake data were small amounts of processing has been done in the form of affine transformation, warping, and color matching [29]. The creators believe that this leads to some deepfakes getting better and others worse. These where selected with the idea that it provides a greater variety of deepfakes.

At this point, the dataset consisted of only deepfakes, and it was important to have real videos in addition for the machine learning models to relate to both types. As previously stated, the test-set from DFDC was used, and is where the real videos used in the custom dataset originates from. In the test-set, labels specifying what were deepfakes and not were available, making it easy to retrieve the real videos. However, since the test-set consisted of a large amount of augmentations, 79% to be exact [8], there was a need to remove the affected videos. But the test-set consists of a total of 5.000 videos, half of which are real videos, which gives 2.500 videos [8]. Of these, 79% are subject to various augmentations, removing them only 525 raw videos without any augmentations remains. This is a bit too small, and the goal was to match the number of real videos with the number of deepfakes, which meant that the number of real videos had to be doubled. As there was no metadata available to explain whether or not a video was affected by augmentations, a manual review had to be done to humanly judge whether a video was affected by augmentations or not. With some discretion and taking into account that some videos with augmentations are needed to reach the goal of 1000 videos, the manual selection was done with the inclusion of videos with small amount of augmentations that where not dominant. This was a somewhat time consuming process that most likely could have been done in a much better way, but at least it resulted in 1033 real videos. Which means that the custom dataset consist of a total of 2033 videos, of which 1000 are deepfakes and 1033 are real videos. Table 4.1 shows an example of how the dataset is structured, with filenames and a label that states whether a video is a deepfake or not, with value 1 being a deepfake and 0 a real video.

In retrospect from the manual selection, videos have been discovered that consisted of various "distractors" such as images, faces and symbols overlays. These videos were not supposed to be part of the dataset, but a little human error in the selection of videos was inevitable. On reflection, it would have provided a more varied and better dataset if some deepfakes from DFDC were included as well.



	filename	actual
995	995_M039.mp4	1
996	996_M039.mp4	1
997	997_W005.mp4	1
998	998_W005.mp4	1
999	999_W005.mp4	1
1000	aarpyivfys.mp4	0
1001	adykfzegpc.mp4	0
1002	aeibsrjfdo.mp4	0
1003	aejcxligwn.mp4	0
1004	aekpwrkywd.mp4	0

Table 4.1: Dataset Structure Example

4.1.2 Dataset Variations

As the purpose of the project was to explore how deepfake detection models performs on data with different image filters and effects, it was important to define which effects might be interesting to test out. The selected filters and effects where as follows;

- No Filter
- Black and White
- Gaussian Blur - Three different levels: (Light, Medium, and High)
- Gaussian Noise
- Mirrored
- Upside Down
- 90 Degree Rotation
- Random Rotation (Between 5 to 45 degrees)
- Lowered Resolution

A separate dataset was created for each of these effects based on the original custom dataset. All effects were created using different libraries in Python. Figure 4.1 shows an example of all the different image filters and effects.

The dataset with no filter is the original custom dataset that was created. Although no effects or filters have been applied to the data, it can be considered the most important experiment as it provides a baseline to compare the other dataset variations against.

In regards to the Lowered Resolution experiment, the videos from DFDC and DeeperForensics have big differences in the resolutions they provide. All of the data in the DFDC dataset where of a higher quality than DeeperForensics. One of the purposes of this experiment was to even out the difference, and study whether this would have an effect on the results. The videos from DFDC where respectively lowered from 1920x1080 to 640x360 pixels and 1280x720 to 480x360 pixels. In DeeperForensics many of the videos where of low quality at 640x360 pixels, but some where also of higher quality at 1280x720 pixels. The videos at 1280x720 where lowered to 640x360 pixels. In essence making the complete custom dataset evenly at a low resolution that is typical for low quality videos on the internet, and is a downscale in resolution that is common when the internet speed is limited.

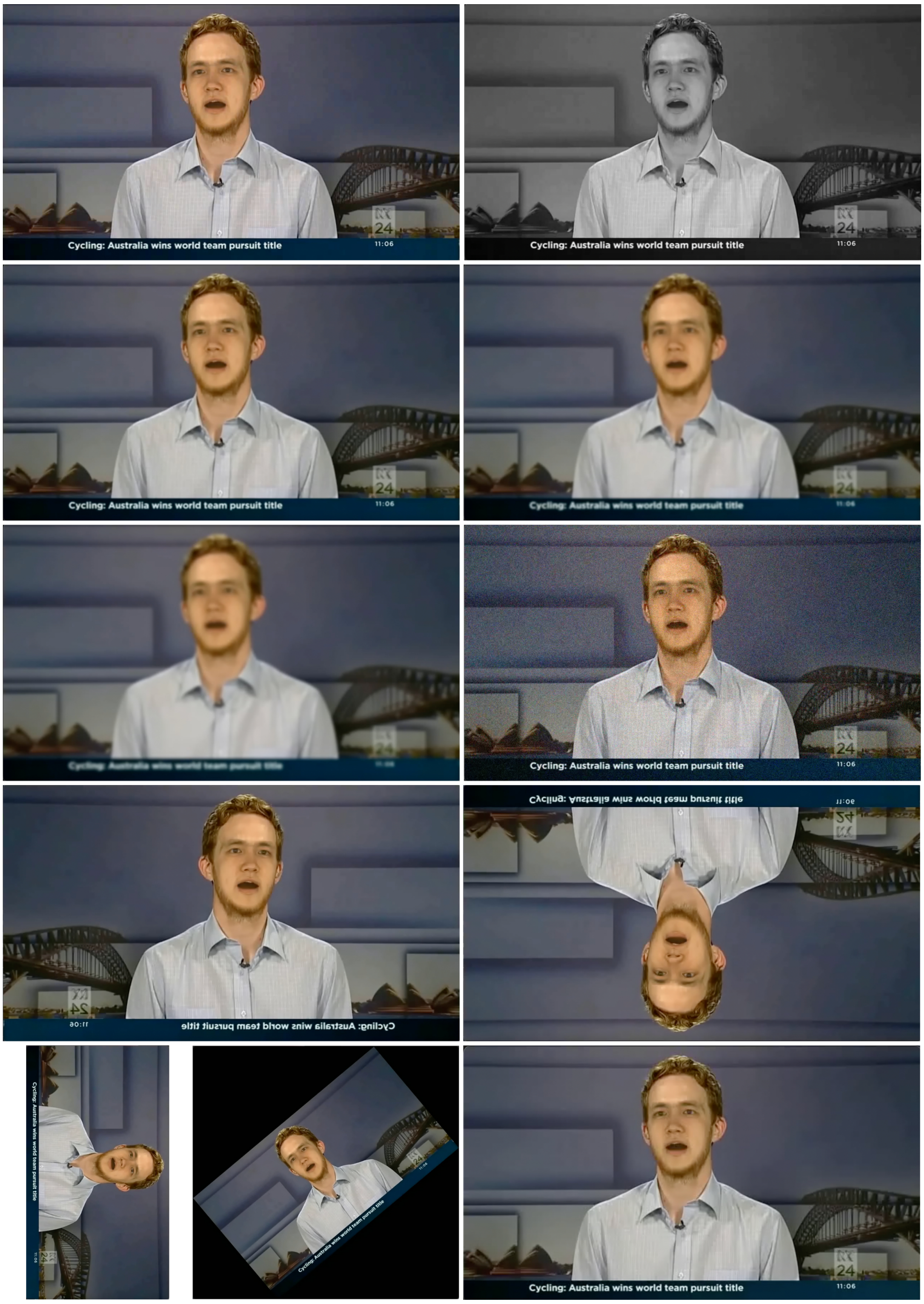


Figure 4.1: Example video of all the different image filters and effects used in the experiments. First row (left to right): No Filter, Black and White. Second Row: Gaussian Blur Light, Gaussian Blur Medium. Third Row: Gaussian Blur High, Gaussian Noise. Fourth row: Mirrored, Upside Down. Fifth row: 90 Degree Rotation, Random Rotation, and Lowered Resolution (From 1280x720 to 640x360).

4.2 Deepfake Detection Models

In order to perform the experiments a deepfake detection model is needed. The choice of model fell on the winner of the DFDC competition, simply because this was the best performing model, in addition to having relatively good documentation on GitHub [34]. Furthermore, it was concluded that it would be beneficial to have an additional model to compare results against, simply because it will be interesting to see if the two models give the same results. Not surprisingly, the choice fell on the second place model in the DFDC competition, which was also publicly available on GitHub [39]. Both models were part of the Deepfake Detection Challenge [24], and both are trained, tested, and validated on the Deepfake Detection Challenge Dataset [8]. It would be interesting to see how the results from these two models are compared to a model trained on other data. Unfortunately, these experiments and testing of the models are a time consuming process, and there was not enough time to test on a third model.

4.3 The Experiments

The execution of experiments and the creation of datasets were both done in parallel, to save time and ensure that there were results available in case there would not be enough time to conduct all the experiments.

4.3.1 Experiment Environment

A proper environment with enough processing power and compatible software was required to run the experiments. Not having access to physical equipment suitable for the task, the choice fell on cloud solutions. Google Cloud, Microsoft Azure, and Amazon Web Services where all considered, but NTNU´s own cloud service SkyHiGh was chosen, simply because it was free and easy to use [40]. This consisted of a virtual machine with Ubuntu Server 20.04 installed and an associated NVIDIA GPU. The deepfake detection models where also cloned from their respective Github repositories [39][34].

4.3.2 Running the Experiments

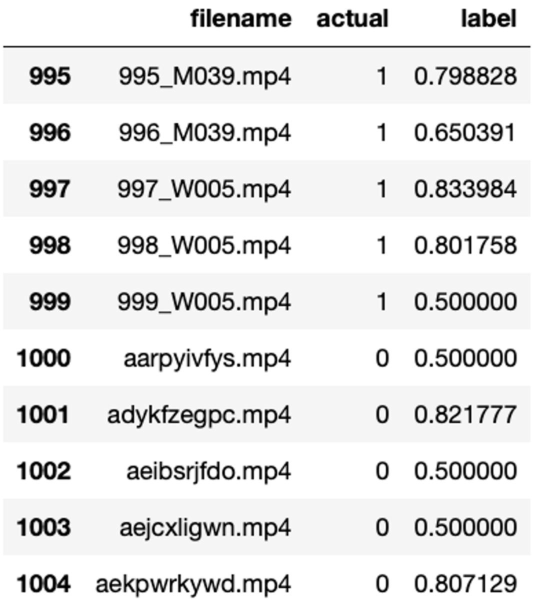
Experiment Output

From the experiments the results are outputted as csv files containing the filenames and label with the predicted probability of the video being a deepfake. The Table 4.2 shows an example output from the 90 Degree predictions on Model Nr.1. From the figure we see that video "995_M039.mp4" is a deepfake and the model is 79.9% certain of it. Whereas the video "adykfzegpc.mp4" is a real video, but the model is 82% certain the video is a deepfake, thus making a wrong prediction.

The video "999_W005.mp4" has a prediction of "0.500000", this value is a prediction error from the model. Table 4.3 and Table 4.4 shows the output from the different experiments on the two models. As can be seen, the total amount of predictions vary a bit on the experiment run on Model Nr.1 as opposed to Model Nr.2. The reason for this is because the amount of prediction errors on Model Nr.1 is higher. The errors are mostly memory related, meaning that there is a possible memory leak in Model Nr.1, yielding out of memory errors.

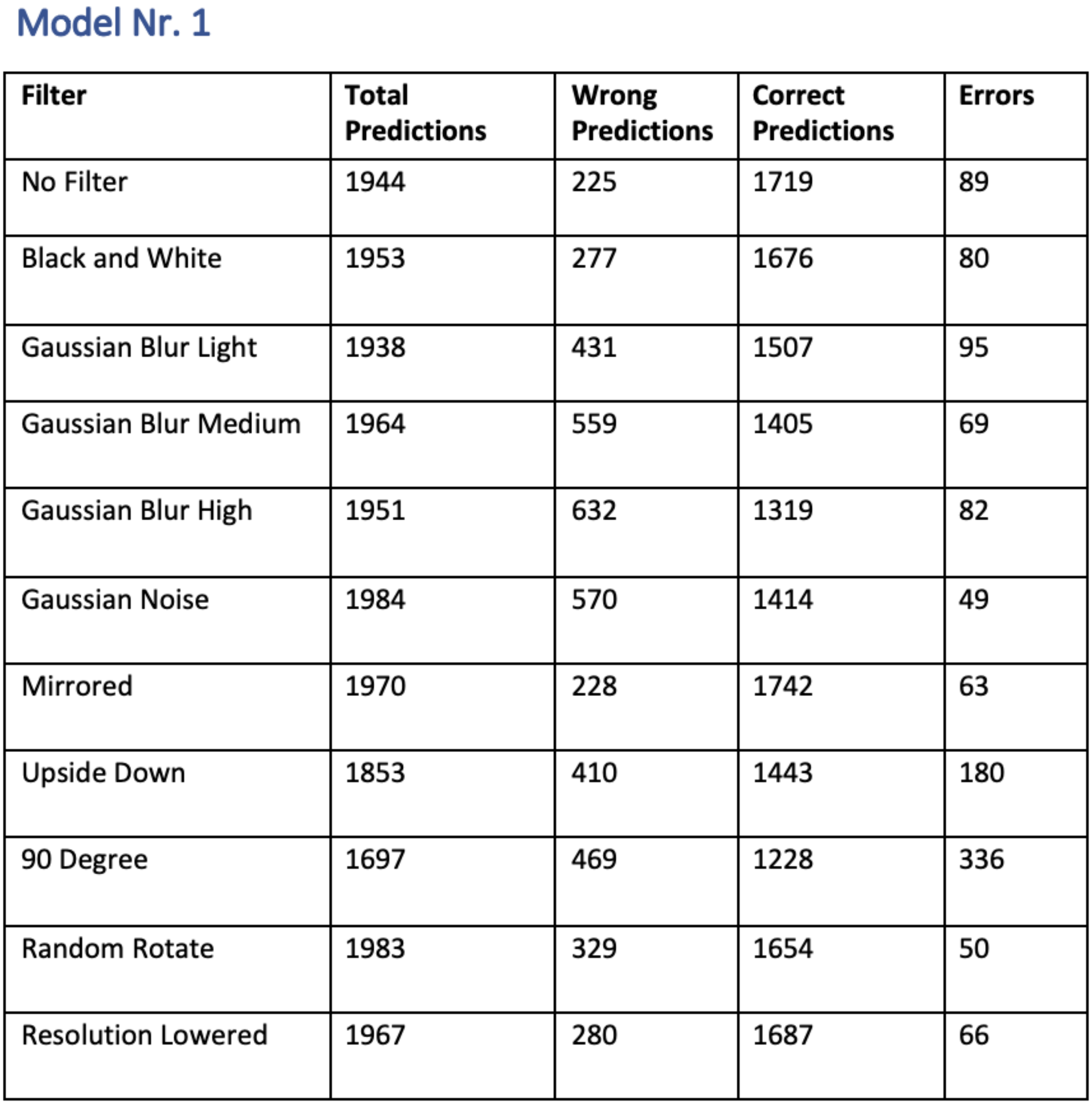
A reduction of the batch size, the amount of videos being predicted in an iteration would probably reduce the amount of errors on Model Nr.1. But would as a result increase the time the model spends for each experiment. As the amount of prediction errors where of such low quantity in conjunction with the total amount of predictions, it was concluded that the number of errors where not critical for the results, and that the time increase for reducing the batch size was not worth it.

From the Table 4.3 one can see that the Upside Down and 90 Degree experiments have more than twice the amount of errors. The reason for this was that they both where prone to unique errors. These two experiments where some of the last to be performed, and there was little to no time to do a throughout review of why the error occurred.



	filename	actual	label
995	995_M039.mp4	1	0.798828
996	996_M039.mp4	1	0.650391
997	997_W005.mp4	1	0.833984
998	998_W005.mp4	1	0.801758
999	999_W005.mp4	1	0.500000
1000	aarpyivfys.mp4	0	0.500000
1001	adykfzegpc.mp4	0	0.821777
1002	aeibsrjfdo.mp4	0	0.500000
1003	aejcxligwn.mp4	0	0.500000
1004	aekpwrkywd.mp4	0	0.807129

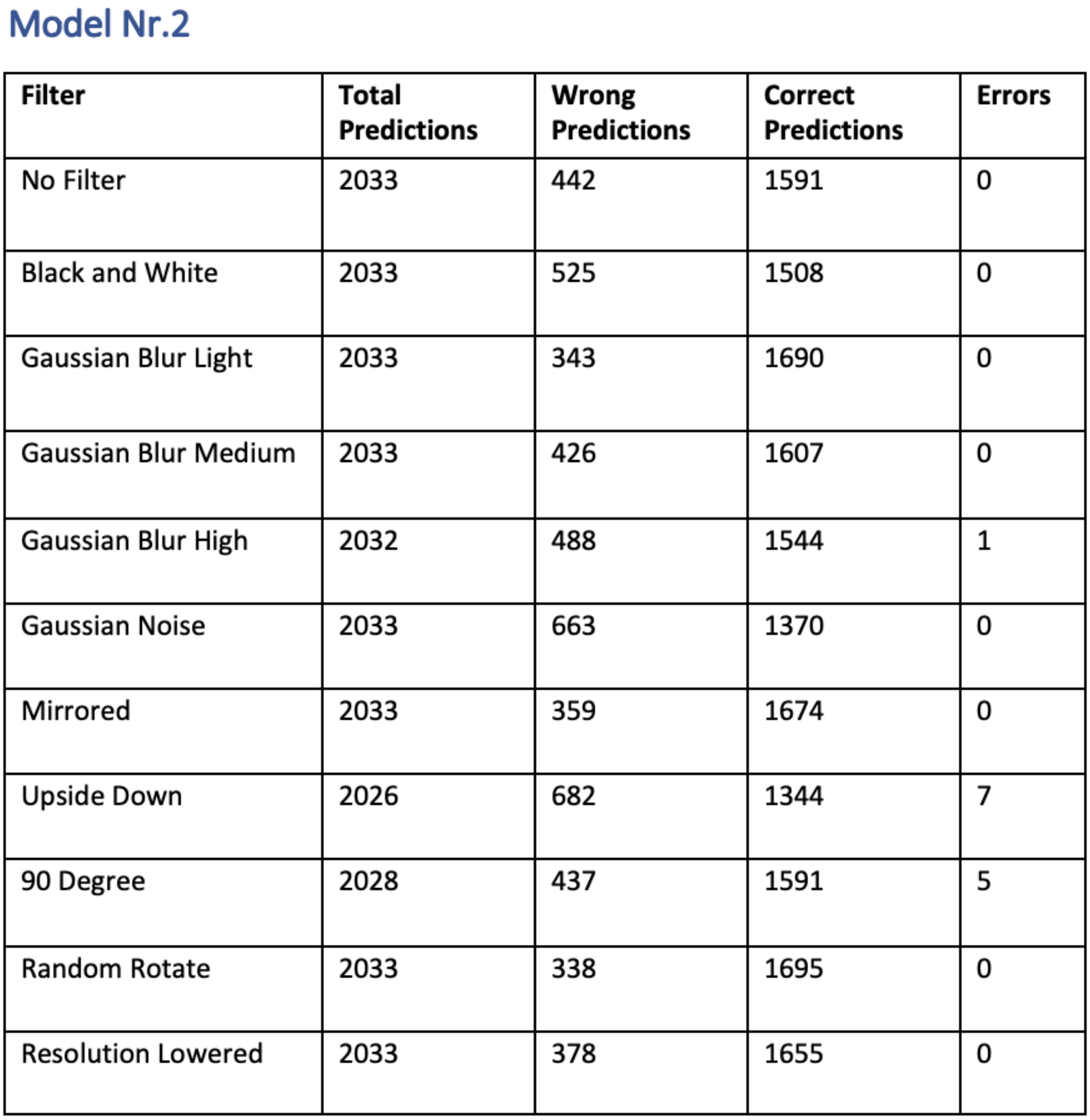
Table 4.2: Example of Predictions from 90 Degree on Model Nr.1



Model Nr. 1

Filter	Total Predictions	Wrong Predictions	Correct Predictions	Errors
No Filter	1944	225	1719	89
Black and White	1953	277	1676	80
Gaussian Blur Light	1938	431	1507	95
Gaussian Blur Medium	1964	559	1405	69
Gaussian Blur High	1951	632	1319	82
Gaussian Noise	1984	570	1414	49
Mirrored	1970	228	1742	63
Upside Down	1853	410	1443	180
90 Degree	1697	469	1228	336
Random Rotate	1983	329	1654	50
Resolution Lowered	1967	280	1687	66

Table 4.3: Model Nr.1 - Experiment Output



Model Nr.2

Filter	Total Predictions	Wrong Predictions	Correct Predictions	Errors
No Filter	2033	442	1591	0
Black and White	2033	525	1508	0
Gaussian Blur Light	2033	343	1690	0
Gaussian Blur Medium	2033	426	1607	0
Gaussian Blur High	2032	488	1544	1
Gaussian Noise	2033	663	1370	0
Mirrored	2033	359	1674	0
Upside Down	2026	682	1344	7
90 Degree	2028	437	1591	5
Random Rotate	2033	338	1695	0
Resolution Lowered	2033	378	1655	0

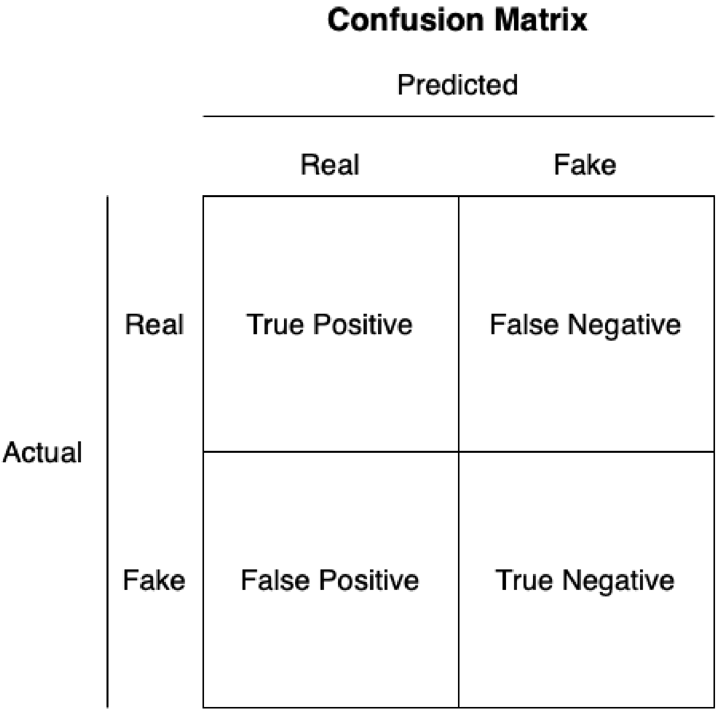
Table 4.4: Model Nr.2 - Experiment Output

4.4 Evaluation Metrics

In order to interpret the results in a meaningful way, relevant and proper metrics are needed. It is important that the choice of metrics are appropriate for the task at hand. With deepfake detection it is essential to measure how accurate the predictions are, both in terms of the percentage of correct predictions and analyze how certain a prediction is. The selected metrics are: **Accuracy**, **Log Loss**, **AUC-ROC (Area Under the Receiver Operating Characteristic Curve)**, **Recall** and **Precision**. A **Confusion Matrix** will also be used to display the results.

4.4.1 Confusion Matrix

The confusion matrix is a table that visually shows the distribution of the classification results [41]. The table is divided into four parts, each representing a different prediction class, **True Positive**, **False Negative**, **False Positive**, and **True Negative**. Figure 4.2 shows an example of a confusion matrix. In the context of the experiments, True Positive represent the number of correctly predicted real videos, and True Negative as correctly predicted deepfakes. Whereas False Negative represent real videos predicted as deepfakes, and False Positives as deepfakes predicted as real videos.



Confusion Matrix

		Predicted	
		Real	Fake
Actual	Real	True Positive	False Negative
	Fake	False Positive	True Negative

Figure 4.2: Example of a Confusion Matrix

4.4.2 Accuracy

Accuracy is a simple metric that outputs how accurate a model is. Meaning that it calculates the percentage of correct predictions. This is calculated by the sum of true positives and true negatives divided by the total number of predictions [42]. Equation (4.1) show the formula for the accuracy metric [42]. With regards to deepfake detection, the true positives are the number of correctly predicted deepfakes while the true negatives are the number of correctly predicted real videos. While accuracy gives a good indication of how good a model is, it does not explain or show how certain the results are. The predicted results are a percentage of how certain the model thinks a video is a deepfake or not. Accuracy treats everything above 50% as a deepfake and everything below as a real video. In addition the amount of deepfake videos in conjunction with real videos will be heavily skewed in a real world scenario, making accuracy a poor metric [8]. This is because if 95% of videos in a set are real videos, and only 5% are deepfakes, a model that predicts everything as real videos will give a accuracy of 95% which is really good. While in reality it cant detect any deepfakes at all, making the model useless.

$$Accuracy = \frac{TruePositive + TrueNegative}{TotalPredictions} \tag{4.1}$$

4.4.3 Recall

Recall is another useful evaluation metric that calculates how many of a specific category the models successfully detects [43]. In this project it will be interesting to see how the different experiments detects deepfakes as opposed to real videos. Recall will therefore be calculated for both deepfakes and real videos, in order to give an understanding of how the models relate to both. Equation (4.2) shows the Recall formula [43].

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{4.2}$$

4.4.4 Precision

Precision is a evaluation metric used to measure how much of a certain prediction is actually correct [43]. This can be used to measure how many of the videos that are predicted as deepfakes actually are deepfakes. Equation (4.3) shows the Precision formula [43].

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{4.3}$$

4.4.5 Log Loss

Log Loss is another popular metric commonly used in classification problems [44]. The metric was used to evaluate the models in the Deepfake Detection Challenge and declare the winner [24]. The use of Log Loss will not only give a good evaluation of the various experiments, but will also provide a basis for comparison against the results from DFDC.

Log Loss is used to evaluate how far off the predictions are from the actual value [44]. Meaning that a correctly predicted deepfake of 90% probability will give a better Log Loss score than a similar prediction with 70% probability. Whereas wrongful predictions will result in a poor Log Loss score. The Log Loss equation is shown in Equation (4.4)[45]. A lower log loss score is better than a high one. In this project log loss is calculated using scikit-learn´s Log Loss metric [45].

$$\begin{aligned} &y = actual \\ &p = predicted \\ &LogLoss = -(ylog(p) + (1-y)log(1-p)) \end{aligned} \tag{4.4}$$

4.4.6 AUC-ROC

AUC-ROC (Area Under the Receiver Operating Characteristic Curve) is a common evaluation metric for classification problems [46]. It is a graph and a score that outputs how good a model is to correctly predict and separate between the two classes [46]. In the example of deepfakes, the AUC-ROC score will assess how good the model is at predicting videos that are deepfakes and not. An AUC-ROC score of 1 means that the model will predict with 100% certainty all deepfakes and all real videos. Whereas a score of 0 means that the model detect all deepfakes as real videos and vice versa. Instead if the AUC-ROC score is 0.5, the model can´t separate between deepfakes and real vidoes, and it only guesses what is what [46]. Figure 5.1 shows an example of the AUC-ROC score for Model Nr.1 and Nr.2 on all the experiments. The AUC-ROC score from the experiment are calculated using scikit-learn´s AUC-ROC score [47].

Chapter 5

Results and Analysis

After successfully performing all the experiments on the two detection models, the data and the results this has given will be very interesting to look at. It will be interesting to see how the different filters and effects are compared to the original custom dataset, in addition to study how the two different models perform in conjunction with each other.

5.1 Experiment Results

The results from Model Nr.1 are shown in Table 5.1, and the results from Model Nr.2 are shown in Table 5.2. The confusion matrices and AUC-ROC graphs for all the different experiments are available in appendix A.

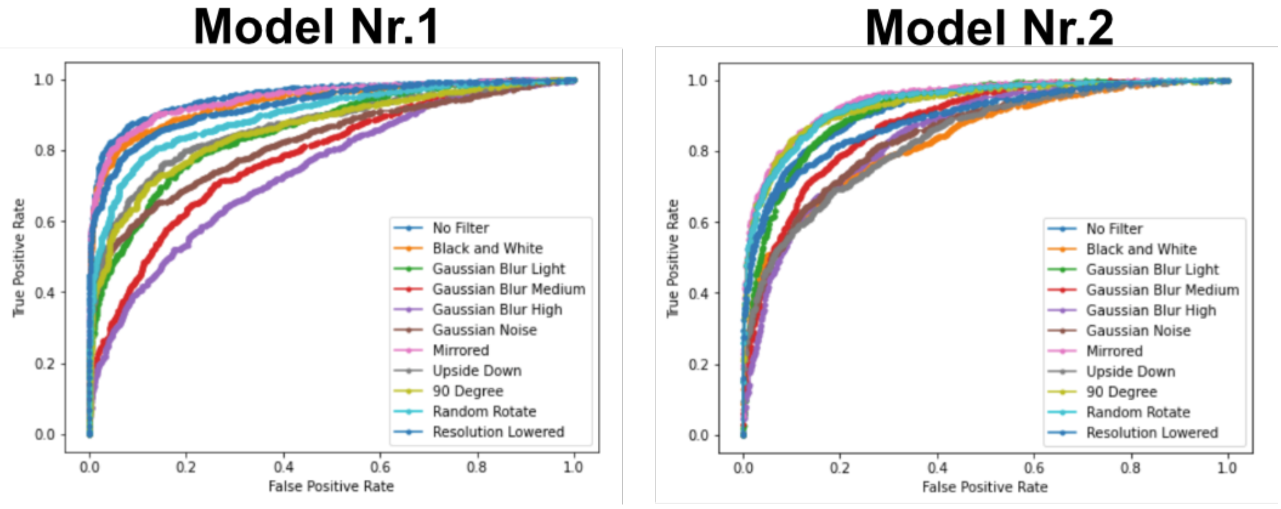
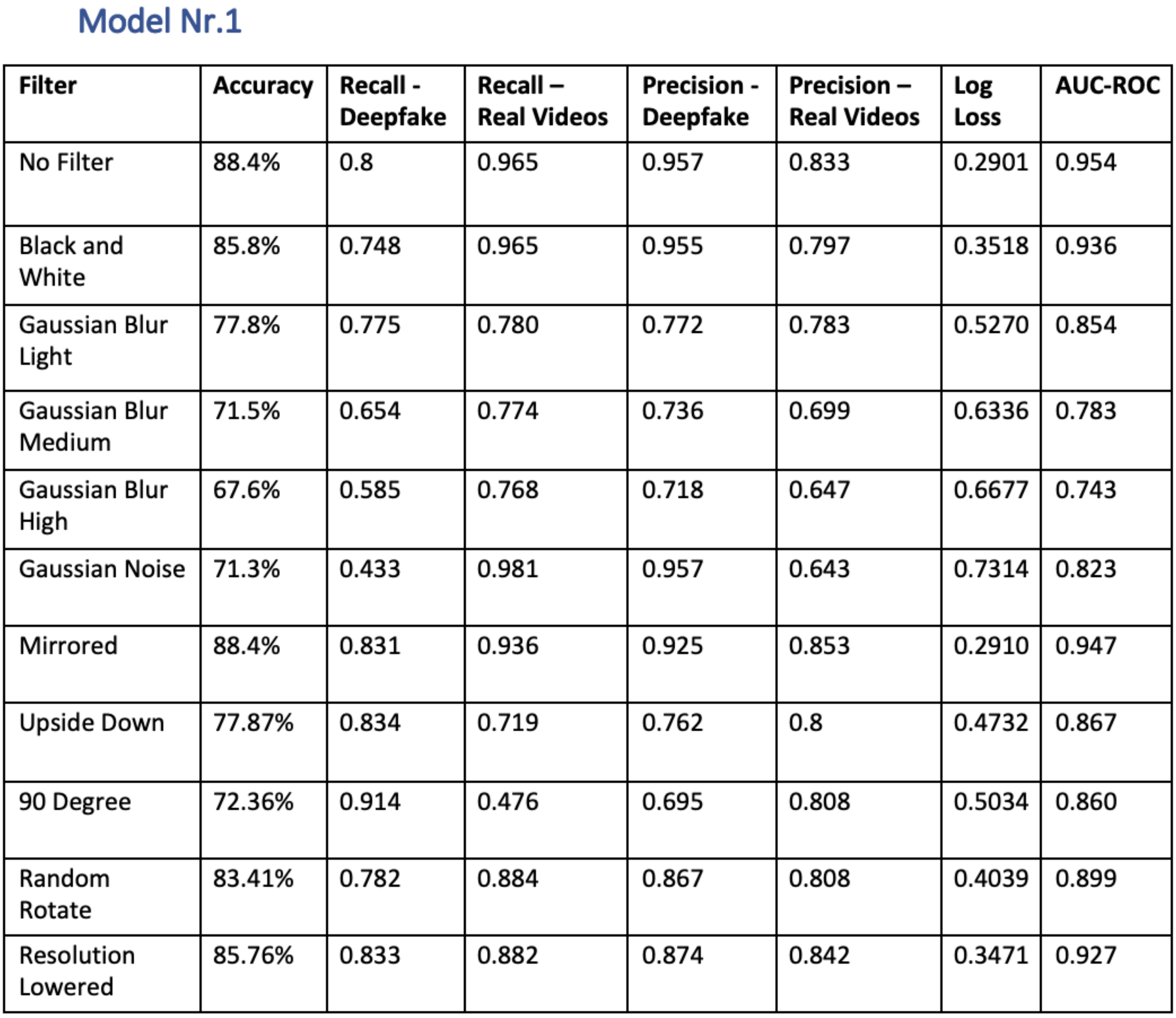


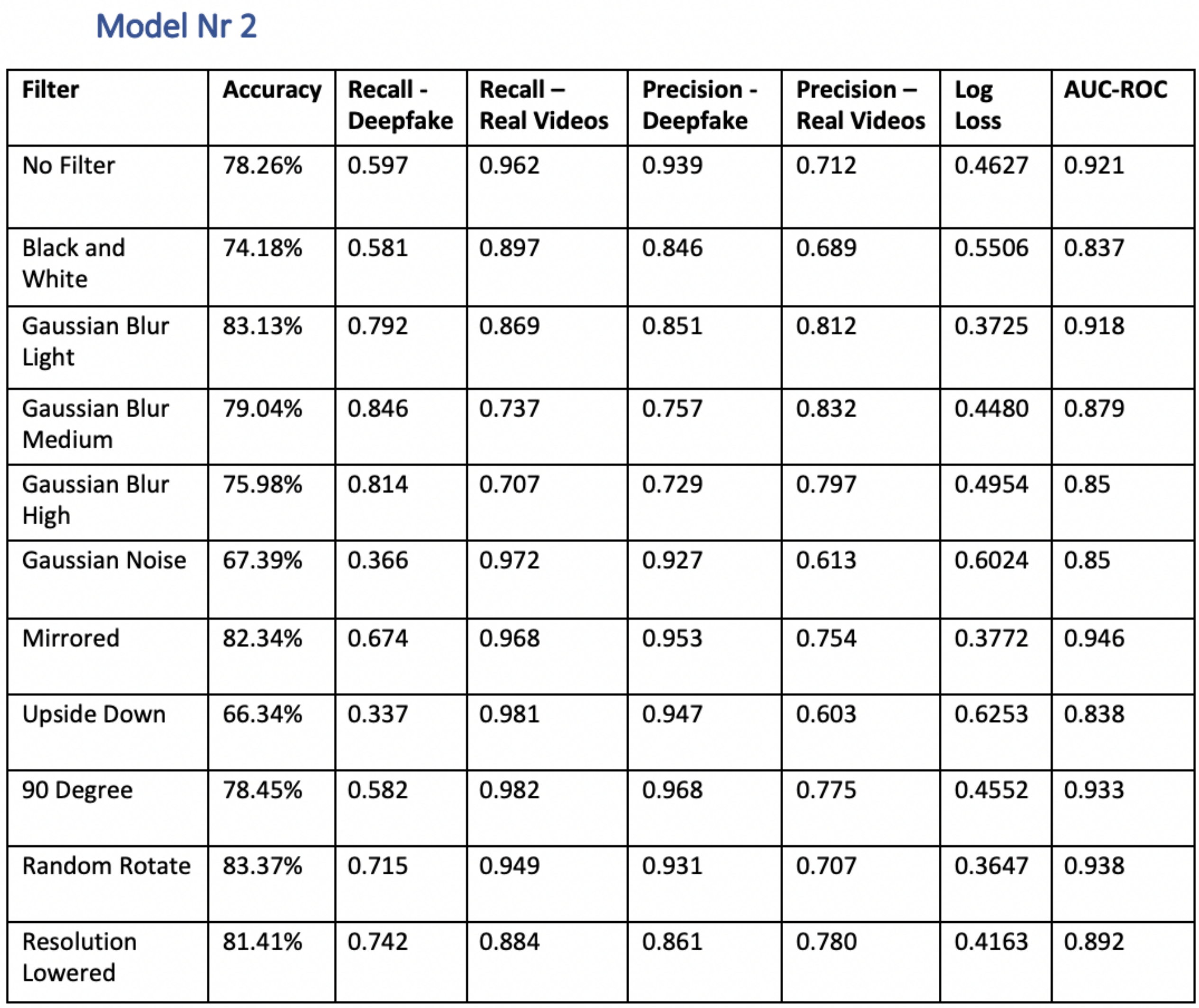
Figure 5.1: AUC-ROC Graph



Model Nr.1

Filter	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
No Filter	88.4%	0.8	0.965	0.957	0.833	0.2901	0.954
Black and White	85.8%	0.748	0.965	0.955	0.797	0.3518	0.936
Gaussian Blur Light	77.8%	0.775	0.780	0.772	0.783	0.5270	0.854
Gaussian Blur Medium	71.5%	0.654	0.774	0.736	0.699	0.6336	0.783
Gaussian Blur High	67.6%	0.585	0.768	0.718	0.647	0.6677	0.743
Gaussian Noise	71.3%	0.433	0.981	0.957	0.643	0.7314	0.823
Mirrored	88.4%	0.831	0.936	0.925	0.853	0.2910	0.947
Upside Down	77.87%	0.834	0.719	0.762	0.8	0.4732	0.867
90 Degree	72.36%	0.914	0.476	0.695	0.808	0.5034	0.860
Random Rotate	83.41%	0.782	0.884	0.867	0.808	0.4039	0.899
Resolution Lowered	85.76%	0.833	0.882	0.874	0.842	0.3471	0.927

Table 5.1: Model Nr.1 Results



Model Nr 2

Filter	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
No Filter	78.26%	0.597	0.962	0.939	0.712	0.4627	0.921
Black and White	74.18%	0.581	0.897	0.846	0.689	0.5506	0.837
Gaussian Blur Light	83.13%	0.792	0.869	0.851	0.812	0.3725	0.918
Gaussian Blur Medium	79.04%	0.846	0.737	0.757	0.832	0.4480	0.879
Gaussian Blur High	75.98%	0.814	0.707	0.729	0.797	0.4954	0.85
Gaussian Noise	67.39%	0.366	0.972	0.927	0.613	0.6024	0.85
Mirrored	82.34%	0.674	0.968	0.953	0.754	0.3772	0.946
Upside Down	66.34%	0.337	0.981	0.947	0.603	0.6253	0.838
90 Degree	78.45%	0.582	0.982	0.968	0.775	0.4552	0.933
Random Rotate	83.37%	0.715	0.949	0.931	0.707	0.3647	0.938
Resolution Lowered	81.41%	0.742	0.884	0.861	0.780	0.4163	0.892

Table 5.2: Model Nr.2 Results

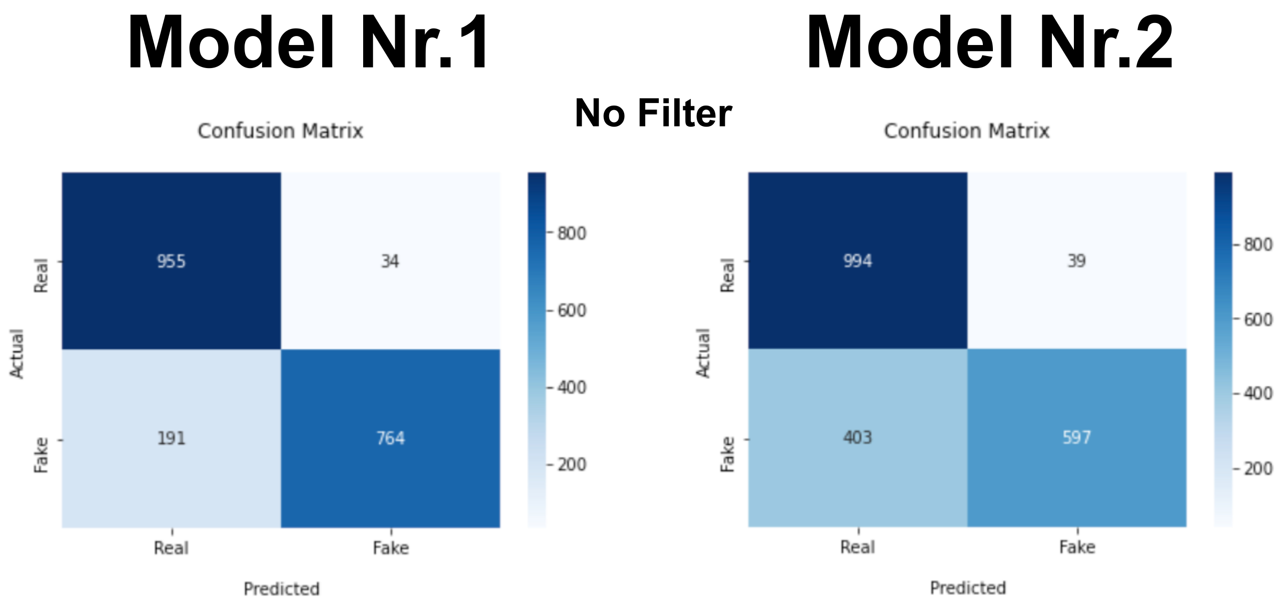


Figure 5.2: Confusion Matrix No Filter

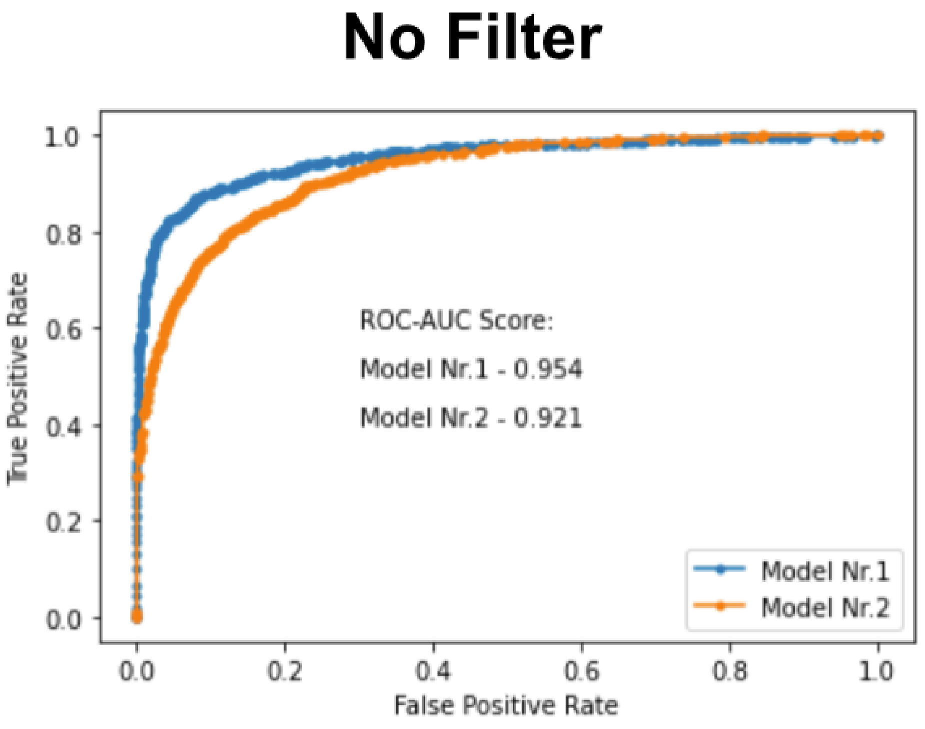
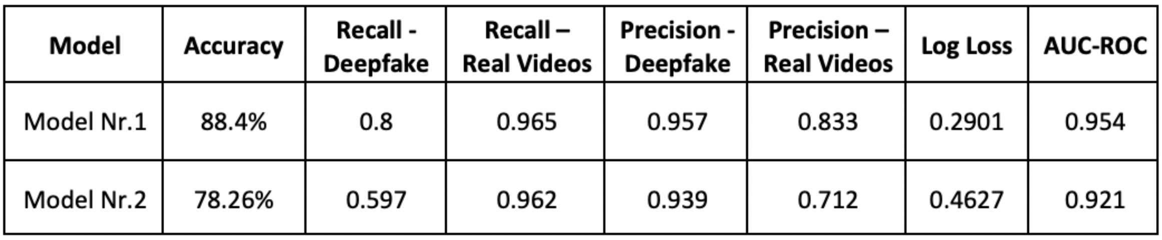


Figure 5.3: AUC-ROC Graph - No Filter

5.1.1 No Filter - Results



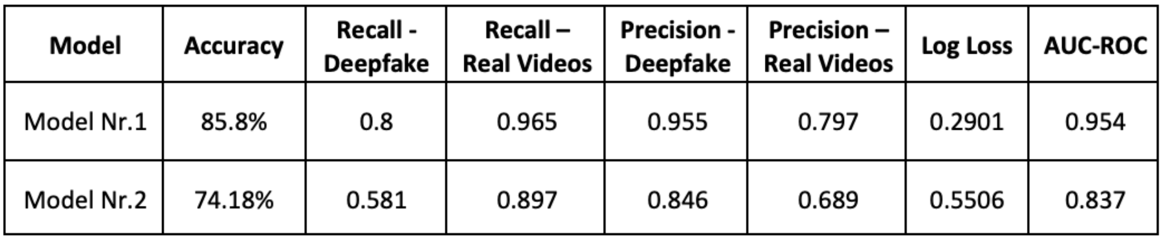
Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	88.4%	0.8	0.965	0.957	0.833	0.2901	0.954
Model Nr.2	78.26%	0.597	0.962	0.939	0.712	0.4627	0.921

Table 5.3: No Filter Results

On Model Nr.1 as expected, No Filter where the best performing experiment overall, achieving 88.4% accuracy and the best log loss and AUC-ROC score.

Surprisingly, the No Filter experiment where far from the best on Model Nr.2, being the the 7th best when looking at accuracy and log loss. Whereas looking at AUC-ROC, No Filter gives the 4th best results. Also the accuracy is 10% lower than the similar experiment on Model Nr.1. Part of the reason why is because that the recall score for deepfakes on Model Nr.2 is considerably lower, 0.597 compared to 0.8 on Model Nr.1. Meaning that both models are equally good at detecting real videos, but Model Nr.2 wrongfully detects a significant amount of deepfakes as real videos. As can also be seen in the precision score for real videos and in the Confusion Matrix Figure 5.2.

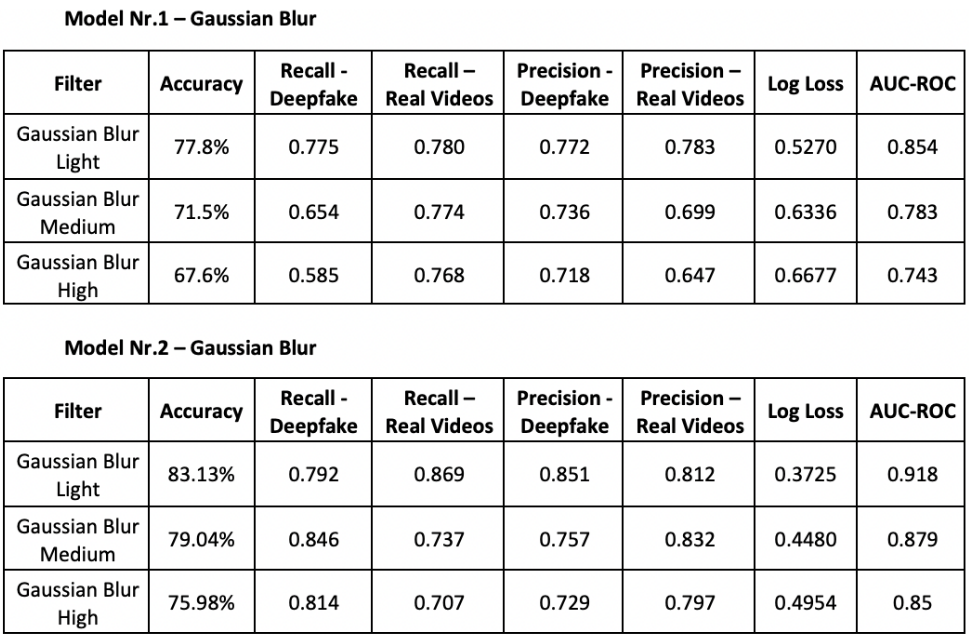
5.1.2 Black and White - Results



Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	85.8%	0.8	0.965	0.955	0.797	0.2901	0.954
Model Nr.2	74.18%	0.581	0.897	0.846	0.689	0.5506	0.837

Table 5.4: Black and White Results

The Black and White experiments perform fairly similar on both models, with a decrease in accuracy of about 3-4% compared to the No Filter results. With regards to log loss and the AUC-ROC score, Model Nr.2 has a slightly higher decrease in score than Model Nr.1. This shows that the Black and White filter has a very small influence on the detection rate.



Model Nr.1 – Gaussian Blur

Filter	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Gaussian Blur Light	77.8%	0.775	0.780	0.772	0.783	0.5270	0.854
Gaussian Blur Medium	71.5%	0.654	0.774	0.736	0.699	0.6336	0.783
Gaussian Blur High	67.6%	0.585	0.768	0.718	0.647	0.6677	0.743

Model Nr.2 – Gaussian Blur

Filter	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Gaussian Blur Light	83.13%	0.792	0.869	0.851	0.812	0.3725	0.918
Gaussian Blur Medium	79.04%	0.846	0.737	0.757	0.832	0.4480	0.879
Gaussian Blur High	75.98%	0.814	0.707	0.729	0.797	0.4954	0.85

Table 5.5: Gaussian Blur Results

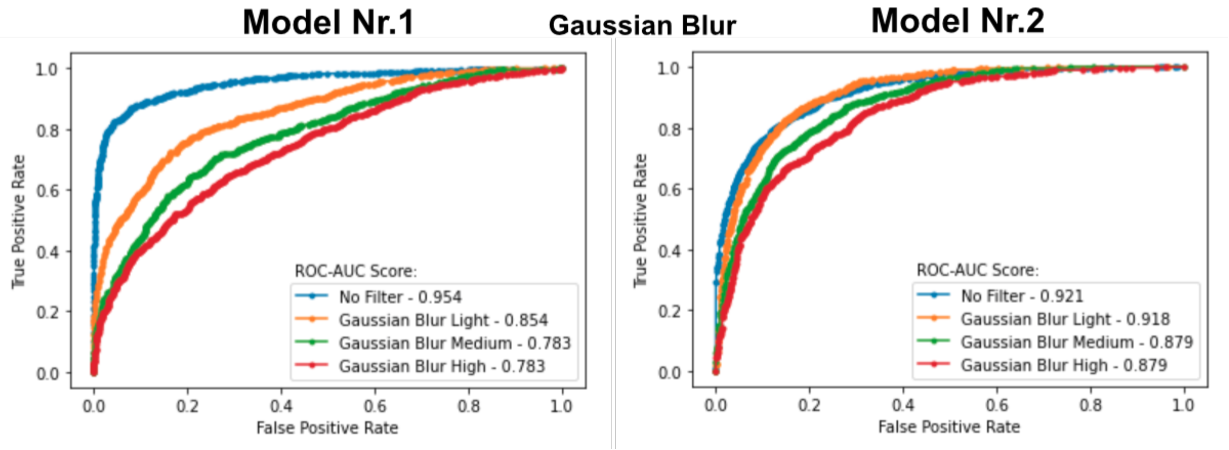


Figure 5.4: AUC-ROC Graph - Gaussian Blur

5.1.3 Gaussian Blur - Results

From the experiments on Model Nr.1 the results are as expected, with all the different blur levels having a significant impact on the results. Whereas the light blur are the least influential, followed by medium and lastly high blur being the most prominent. From the recall, all the three levels of blur performs fairly similarly on real videos, ranging from 0.780 on light to 0.768 on high. Whereas the recall values on deepfakes have substantially greater influence, ranging from 0.775 on light blur to 0.585 on the high blur. This tells us that applying blur makes it significantly harder for the model to detect deepfakes. A detail and flaw in the data that can explain the big difference in the recall value for deepfakes and real videos, is the quality and resolution difference in the videos. As the real videos generally have a much higher quality and resolution than the deepfake videos in the custom dataset, a theory is that the appliance of blur have a greater impact on videos with lower resolution. Figure 5.5 shows two videos from the Gaussian Blur High dataset, a deepfake and a real video. The deepfake is one of the videos in the dataset with the lowest resolution and the real video is one of the videos with the highest resolution. Both of these videos have been applied the same level of blur, but the deepfake with lower resolution is clearly more blurry. Confirming the theory that the lower resolution videos are more affected by the blur.

Another observation is that the appliance of light blur reduces the recall value of real videos to a greater extent than deepfakes when comparing to the No Filter experiment. The recall on real videos drops from 0.965 on No Filter to 0.780 on the light blur, whereas the deepfake recall only drops from 0.8 to 0.775. Another theory is that the appliance of blur have a general impact on the real videos, but the amount of blur applied have a much greater significance on the deepfake videos.

With regards to Model Nr.2 the results are completely different. Here the appliance of light and medium blur increases the accuracy and log loss compared to the No Filter experiment, something quite the opposite of the hypothesis. Even though the accuracy increases, the AUC-ROC score decreases slightly, this might be because there is more uncertainty in the results it provides. Example the model might predict a video with 90% certainty as a real videos which is correct, but with the added blur the certainty decreases to 70%. While also correct, the model is less certain of the result.

Studying the recall values for Model Nr.2 there is a better understanding for the increased blur results. The recall value of deepfakes in the experiments with No Filter is very low, at only 0.597. While the deepfake recall in the blur experiments ranges from 0.792 to 0.846, with medium blur being the highest. This shows that the appliance of blur increases the detection rate of deepfakes for the model. The recall for real videos on the other hand decreases when compared to the No

Filter experiment. Here the No Filter is 0.962 and the blur experiment range from 0.869 to 0.707, with light blur giving the best result and the high blur the worst. All things considered, the appliance of blur slightly decreases the detection of real videos but increases the detection rate of deepfakes. This can be related to the resolution distribution of deepfakes and real videos, where the blur is more prominent on the low resolution deepfakes. In relation to the result a theory is that Model Nr.2 is trained to detect blurriness as deepfakes, therefore applying blur increases the amount of deepfake detected.

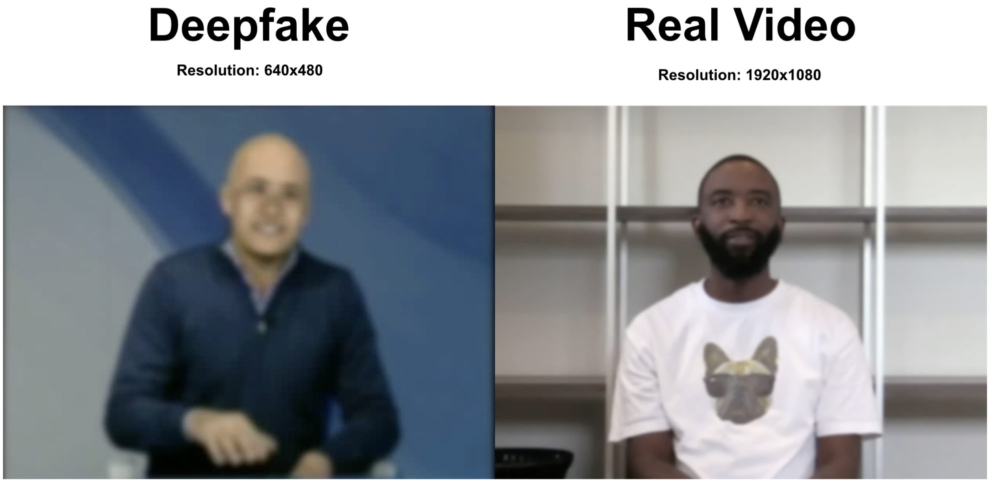
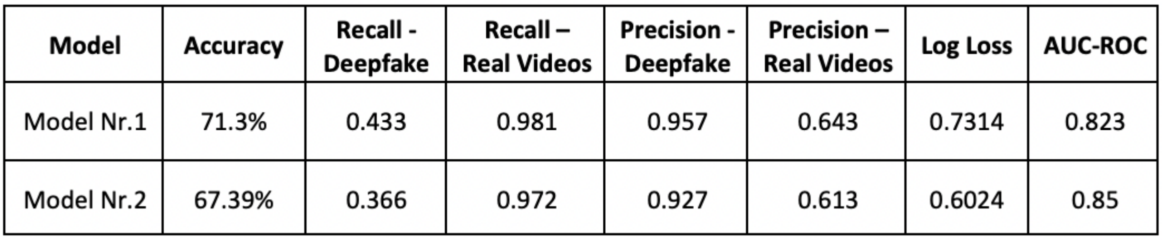


Figure 5.5: Example of high level blur on video with low resolution and high resolution. Note: Part of the background in the images have been cropped out to better illustrate the blur effect on the faces. Unedited, the deepfake is originally part of the DeeperForensics dataset [9] and the real video is originally part of the DFDC dataset [8].

5.1.4 Gaussian Noise - Results



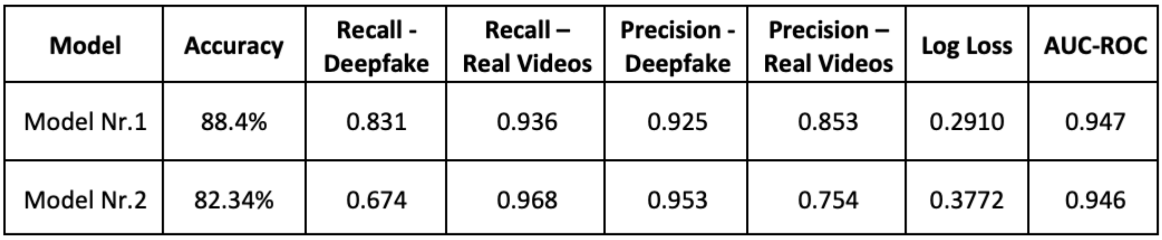
Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	71.3%	0.433	0.981	0.957	0.643	0.7314	0.823
Model Nr.2	67.39%	0.366	0.972	0.927	0.613	0.6024	0.85

Table 5.6: Gaussian Noise Results

The Gaussian Noise experiments also provide some interesting results. Here the results from both models are fairly similar, with only 4% difference in accuracy, 71.3% on Model Nr.1 and 67.39% on Model Nr.2, and the AUC-ROC score being almost identical. The Log Loss on the other hand is also interesting, as Model Nr.2 gives the best score of 0.6, whereas Model Nr.1 is at 0.73. In relation to accuracy and log loss this may mean that Model Nr.1 is predicting less videos wrong than Model Nr.2, but the wrongfully predictions is to a greater extent wrong, which negatively influence the log loss score to a greater degree.

Something very interesting in this experiment is how the recall values are affected. The difference between the recall values on deepfakes and real videos is great. Both models perform similar on real videos, with a recall value of 0.981 on Model Nr.1 and 0.972 on Model Nr.2. The recall value on deepfake on the other hand is 0.433 on Model Nr.1 and 0.366 on Model Nr.2. In comparison to the No Filter experiments, the appliance of noise does not seem to affect the recall value of real videos, but have a huge impact on the recall value of deepfakes. By looking at Model Nr.1 the recall value of deepfake is almost halved, from 0.8 on No Filter to 0.433 on the Gaussian Noise experiment. Considering Model Nr.2, the decrease is fairly similar, almost halved from 0.597 on No Filter to 0.337 on the Gaussian Noise. Looking at the Confusion Matrix in Appendix A and the precision scores, we see that both models predict less videos as deepfakes. The precision score for deepfakes is very high, meaning that most of the predicted deepfakes are correctly classified. This shows that the appliance of noise results in both models predicts most videos as real videos, thus a lot of deepfake videos are wrongfully classified as real.

5.1.5 Mirrored - Results

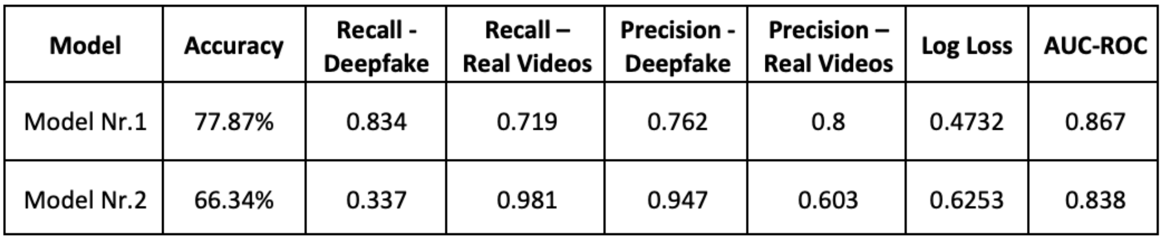


Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	88.4%	0.831	0.936	0.925	0.853	0.2910	0.947
Model Nr.2	82.34%	0.674	0.968	0.953	0.754	0.3772	0.946

Table 5.7: Mirrored Results

Not surprisingly the mirrored experiments does not have any big influence on the results. Compared to the basic No Filter experiments, the results on Model Nr.1 is practically identical. On Model Nr.2, the mirrored results compared to No Filter is slightly more different, with the mirrored experiment having 4% better accuracy. The difference is very small, and there is possible that it is only a small coincidence that influence the result.

5.1.6 Upside Down - Results



Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	77.87%	0.834	0.719	0.762	0.8	0.4732	0.867
Model Nr.2	66.34%	0.337	0.981	0.947	0.603	0.6253	0.838

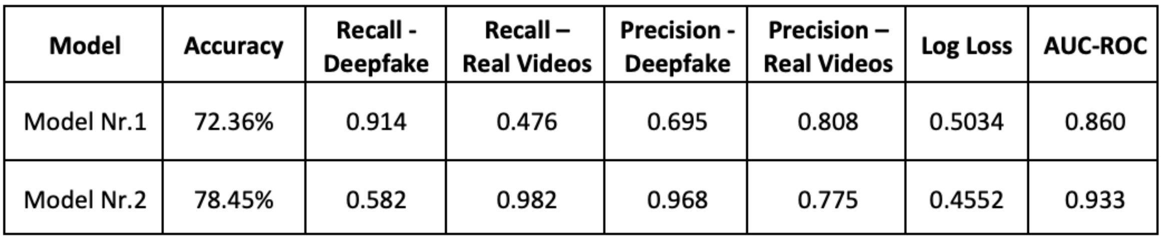
Table 5.8: Upside Down Results

The Upside Down experiment gives some interesting results, one would probably think that turning a video upside down would not have a significant affect on the result.

On Model Nr.1 there is a 10% decrease in accuracy compared to the basic No Filter experiment at 77.87%. The log loss and AUC-ROC score also have a slight decrease. By looking at the recall, we can see the reason for the decrease in result. The recall on deepfakes is basically the same as for No Filter at 0.834, whereas the recall for real videos have some reduction, from 0.965 with No Filter to 0.719 on the Upside Down.

Whereas on Model Nr.2 the decrease is practically similar. The accuracy score is at 66.34%, which is a 12% decrease when compared to No Filter. Interestingly as opposed to Model Nr.1, the recall on real videos is unchanged whereas the recall on deepfakes is almost halved, from 0.597 on No Filter to 0.337 on Upside Down. Studying the Confusion Matrix in Appendix A and the precision score, we see that Model Nr.2 predicts most videos as real, resulting in a lot of deepfakes being wrongfully classified. There is no good explanation of why the recall results from models are opposite, but the models handles upside down videos differently.

5.1.7 90 Degree - Results



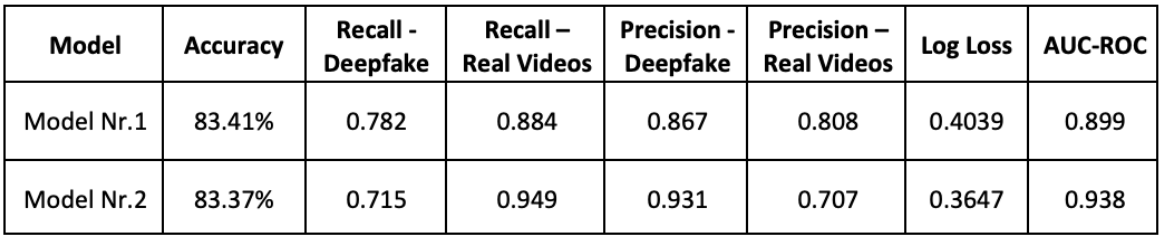
Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	72.36%	0.914	0.476	0.695	0.808	0.5034	0.860
Model Nr.2	78.45%	0.582	0.982	0.968	0.775	0.4552	0.933

Table 5.9: 90 Degree Results

The 90 Degree experiments has some interesting differences between the two models. On Model Nr.1 the accuracy has dropped from 88.4% to 72.36%, and the recall value for deepfakes has increased from 0.8 to 0.914. Whereas the recall on videos have decreased from 0.965 to 0.476. With regards to the Confusion Matrix in Appendix A one can see that Model Nr.1 predicts most videos as deepfakes. It must also be noted that the 90 Degree experiment on Model Nr.1 had the most prediction errors as seen in Table 4.3. Meaning that the experiment contains about 12% less predictions than the basic No Filter experiment. It is not certain that the lower amount of predictions have any effect on the result, but it should be mentioned.

Model Nr.2 on the other hand, performs identical to the No Filter experiment. This shows that Model Nr.1 is the most prone to 90 degree rotation.

5.1.8 Random Rotation - Results



Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	83.41%	0.782	0.884	0.867	0.808	0.4039	0.899
Model Nr.2	83.37%	0.715	0.949	0.931	0.707	0.3647	0.938

Table 5.10: Random Rotate Results

Random rotation is the last of the rotated experiments and provide some interesting results. With regards to Model Nr.1 the results are as expected with a slight decrease in accuracy, Log Loss and AUC-ROC. Looking at the recall, the model detects deepfakes equally as No Filter, but have a decrease in the detection of real videos.

Model Nr.2 on the other hand surprisingly performs better than the No Filter experiment, with a 5% increase in accuracy, from 78.26% to 83.37%. By studying the recall values, the model predicts equally on real videos, but is better at correctly predicting deepfakes than the the result from No Filter.

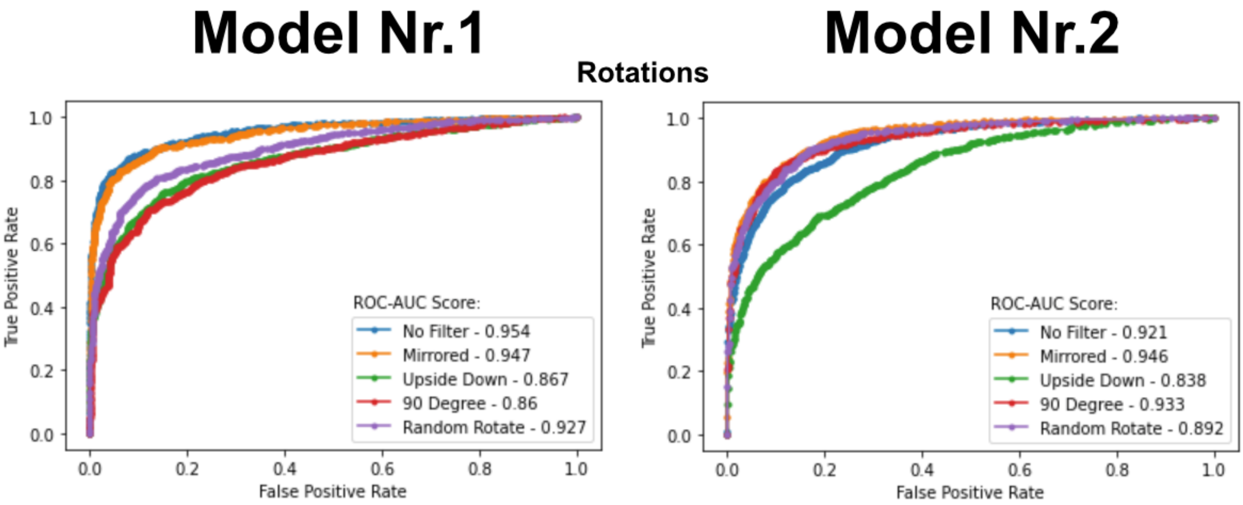
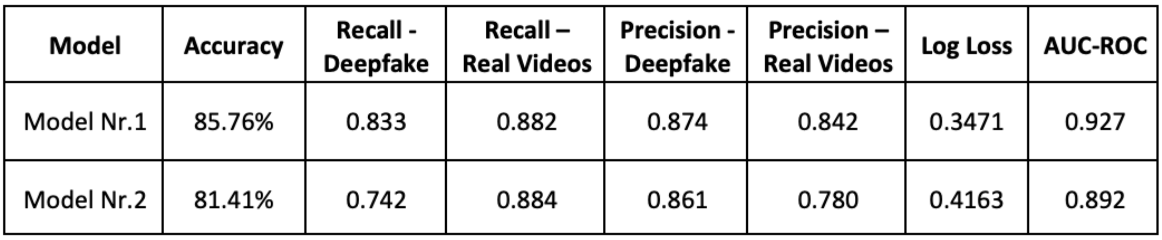


Figure 5.6: AUC-ROC Graph - Rotations

5.1.9 Resolution Lowered - Results



Model	Accuracy	Recall - Deepfake	Recall – Real Videos	Precision - Deepfake	Precision – Real Videos	Log Loss	AUC-ROC
Model Nr.1	85.76%	0.833	0.882	0.874	0.842	0.3471	0.927
Model Nr.2	81.41%	0.742	0.884	0.861	0.780	0.4163	0.892

Table 5.11: Resolution Lowered Results

The results from lowering the resolution gives insight into how the models adapt to videos of lower quality.
On Model Nr.1 the results are fairly similar with a small increase in accuracy from 88.4% to 85.76%. The recall value on deepfakes is almost similar, whereas on real videos there is a small decrease from 0.965 to 0.882.

In regards to Model Nr.2 the results are actually a little bit better, with an increase in accuracy from 78.26% to 81.41%. Whereas the AUC-ROC score actually have a slight decrease. Studying the recall value it is noticed that there is a small decrease in the detection rate of real videos, but there is an increase in the detection of deepfakes.

The results shows that lowering the resolution does not have a big influence on the results, but both models have a small decrease in the detection of real videos. This could be because the real videos have had their resolution lowered the most.

5.2 Deepfake Detection Challenge Result Comparison

Having completed all the experiments with corresponding results, it will be interesting to see how this project compare to the results from the Deepfake Detection Challenge - DFDC.

The best performing model from DFDC scored 82.56% accurate on the public leaderboard, and 65.18% when run on the private test-set [31]. The private test-set is as previously mentioned the origin of the real videos used in this projects custom dataset.
The log loss score for Model Nr.1 ended at 0.42798 on the private test-set and Model Nr.2 scored 0.42842 [31]. Compared to the results from this project, both models performs better in terms of accuracy with No Filter, whereas Model Nr.1 scores 88.4% and Model Nr.2 at 78.26%. With regards to log loss, Model Nr.1 performs better in this project with a log loss of 0.2901 on No Filter compared to

0.42798 from DFDC. This shows that the model is better at generalizing to the custom dataset from this project than the private test-set in DFDC.
Model Nr.2 on the other hand performs slightly worse in this project with a log loss of 0.4627 compared to 0.42842 in DFDC [8]. But the best performing experiment on Model Nr.2 in this project actually performs better than the result from DFDC, with the Random Rotation experiment resulting in a log loss of 0.3647.

To interpret these results, it may be interesting to take a closer look at the private test-set. About 79% of the videos in the private test-set where applied various augmentations as explained in Section 2.2.1 [8]. Additionally, the private test-set consisted of 5.000 videos taken from the internet consisting of real videos and deepfakes, those where removed before the dataset was published. From the analysis of the result in DFDC it has been calculated how the models performs on the DFDC data compared to the videos taken from the internet [8], both from the private test-set. Here Model Nr.1 results in a log loss of 0.1983 on the DFDC part, and 0.6605 on the internet videos [8]. Whereas Model Nr.2 result in a log loss of 0.1787 on DFDC and 0.6805 on the internet videos. This shows that both models generalize better to the DFDC data.

Based on this analysis and information it can be concluded that Model Nr.1 performs and generalize better on the DeeperForensics data than the internet videos from the DFDC private test-set. A difference to keep in mind is that the DeeperForensics data used in this project consist only of deepfakes, but the internet data of DFDC consist of both deepfakes and real videos.

The results from Model Nr.2 on the other hand performs fairly similar on both the basic experiment from this project and the DFDC private test-set. Based on this, it can be concluded that Model Nr.1 is better at generalizing than Model Nr.2.

5.2.1 DFDC Augmentations Result and Comparison

In the DFDC dataset it has as mentioned in Section 2.2.1 been used various augmentations as filters and effects in the validation and test set [8]. Some of these augmentations are similar to those used in this project. The creators have done a small analysis of the result they provide, and Table 5.12 shows the results from the similar augmentations in DFDC. It has not been specified which model is used to output the results, thereby the value the comparison offers will be limited. But it will still be interesting to see if there are similarities in the results.

No Filter / No Augmentation Comparison

The basic result with no augmentation from DFDC scores lower than the No Filter experiments on both models in this project. Showing that the results in this project performs better.

Black and White Comparison

In the DFDC analysis the black and white results in a score that is fairly similar to the no augmentation results. Both models in this project have a log loss score that is a little bit lower than the basis, but both still performs better than DFDC. It should be noted that DFDC have a lower decrease than both models in this project.

Gaussian Blur Comparison

In this project, three different experiments have been done with three levels of blur. The Gaussian blur augmentation in DFDC is applied at random levels making it hard to compare to one of the specific blur experiments in this project.

With regards to Model Nr.1 both the medium and high level blur experiments performs worse than the blur in the DFDC augmentation. The light blur on the other hand have a log loss that is fairly similar to the results from DFDC. Model NR.2 performs better than DFDC in all of the blur levels with the most similar being the high blur.

A detail to keep in mind is that the difference between the basic no augmentation results and the blur is much lower in the DFDC analysis than in the experiments from this project.

Gaussian Noise Comparison

In DFDC the noise augmentation is performing the worst in the analysis, something that is equal on Model Nr.1, and is the second worse result on Model Nr.2. Interestingly the Gaussian Noise experiment on Model Nr.1 performs worse than the results from DFDC. Meanwhile Model Nr.2 performs sligthly better

Horizontal Flipping (Mirroring) Comparison

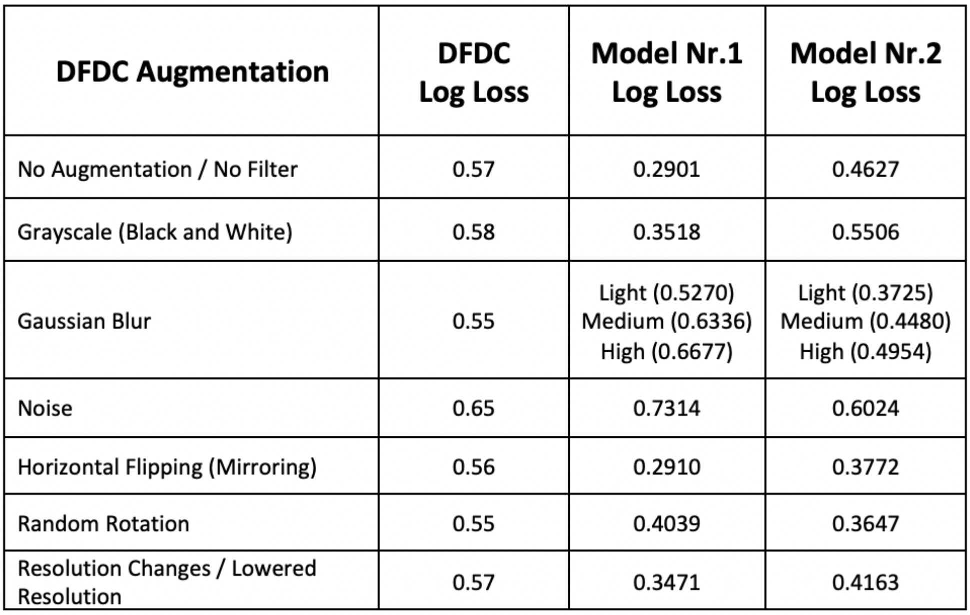
The results from mirroring a video is almost similar to the no augmentation results in the DFDC analysis. The same can be said for Model Nr.1 where mirroring a video gives about the same results.
Model Nr.2 is as previously mentioned performing better than No Filter on the mirrored experiment.

Random Rotation Comparison

The rotation in DFDC is applied at random, but the range it is rotated is not specified. With regards to the results, both models experiments in this project perform better than DFDC. But as have been the case with multiple other augmentation, is that DFDC have a log loss decrease that is lower than the experiments in this project.

Resolution Changes Comparison

The resolution changes in DFDC is applied at random, but the range of different resolution changes is not specified. In DFDC the resolution augmentation perform equal to the no augmentation result. Both models in this project perform better than DFDC, but the log loss change is higher when compared to the No Filter experiment.



DFDC Augmentation	DFDC Log Loss	Model Nr.1 Log Loss	Model Nr.2 Log Loss
No Augmentation / No Filter	0.57	0.2901	0.4627
Grayscale (Black and White)	0.58	0.3518	0.5506
Gaussian Blur	0.55	Light (0.5270) Medium (0.6336) High (0.6677)	Light (0.3725) Medium (0.4480) High (0.4954)
Noise	0.65	0.7314	0.6024
Horizontal Flipping (Mirroring)	0.56	0.2910	0.3772
Random Rotation	0.55	0.4039	0.3647
Resolution Changes / Lowered Resolution	0.57	0.3471	0.4163

Table 5.12: Augmentation Results from the Deepfake Detection Challenge (DFDC) [8] and results from similar experiments in this project. The results from DFDC have been manually read from a graph and the numbers are not exact, but an approximation. Note: The DFDC paper does not specify which model is used to produce the results.

Chapter 6

Discussion

6.1 Result Findings

This project and the experiments have led to a number of interesting results that will be exciting to discuss further. Based on the experiments, many of the results were similar to the hypothesis, but at the same time some of them were surprising. The results from the two models were not consistent and there were some irregularities from various experiments. In fact it was the results from Model Nr.1 that was the most similar to the hypotheses, and it was overall the best performer.

6.1.1 Model Nr.1 Findings

Model Nr.1 performed great on the experiments with generally good results. Not surprisingly the basic No Filter experiment was the best performer, followed closely by mirrored, black and white, and the lowered resolution experiment. Results that where mostly as expected. The highest level of blur were the lowest performing experiment on Model Nr.1 followed by Gaussian Noise. The Gaussian Noise was more influential than expected. The upside down and 90 degree experiments also resulted in lower scores. The reason for this is probably as explained in Section 3.1.1, as CNNs are prone to rotation if not trained on them [38].

Generally, Model Nr.1 was better at predicting real videos as opposed to deepfakes. This may be related to the model being trained on similar data as the real videos. The real videos and training data for the models are from the same dataset, but the models have not been trained on the real videos used in this project.

6.1.2 Model Nr.2 Findings

As opposed to the first model, Model Nr.2 consisted of results that was more unexpected. First of all the overall result from the basic No Filter experiment was significantly lower than Model Nr.1, and it was also the 7th ranked experiment accuracy wise on Model Nr.2. This was unexpected as it was expected to perform the best among the experiments with great certainty.

The most surprising thing about the results was that the light and medium blur experiments performed better than the experiment with no filter. Something quite the opposite of Model Nr.1.

An interesting detail is that the upside down and 90 degree experiment from Model Nr.2 was good at predicting real videos. Whereas the same experiments on Model Nr.1 was best at predicting deepfakes. Showing that the two models are affected quite differently on these two rotations.

Otherwise the random rotation, mirrored, lowered resolution and 90 degree also performed better than No Filter on Model Nr.2. The recurring detail is that all of them except 90 degree performs better than No Filter with regards to correctly predicting deepfakes. In addition all of the blur experiments also are better at detecting deepfakes. The only experiments on Model Nr.2 that is significantly worse at detecting deepfakes is the Gaussian Noise and Upside Down experiments.

Both models are quite similar in the detection of real videos, most likely as explained earlier because of the origin of the data. Noticeably, Model Nr.2 is much worse at correctly detection deepfakes, which is a sign of Model Nr.1 being better at generalizing to new data.

6.2 Limitations and Considerations

It must be mentioned that some of the rotation experiments are quite unrealistic to encounter in real world deepfake detection. Especially upside down as there is little point in publishing a video that is upside down. The same thing can partially be said about 90 degree rotation, but due to the mobile phone many videos are filmed at a different angle which makes 90 degree rotation a bit relevant anyway. The usefulness of the random rotation experiment can also be considered, but it is common with a naturally rotated face in videos.

It would also be interesting to test and see how the models performs on data that is not part of the Deepfake Detection Challenge dataset. In this project that would be to replace the real videos with other data, which would give a better picture on how well the models generalize.

Chapter 7

Conclusion

In this thesis the various experiments have contributed with new results and information in regards to deepfake detection. The different experiments have given an indication and answer to how the state of the art deepfake detection models generalize to new data applied various image filter and effects. The results also show the influence the different filters and effects have on the detection rate.

The first research question is about finding out how the deepfake detection models generalize to new data with different image filters and effects. Both models generalize quite well on new modified data, but the results varies gradually based on the filter and effect used. There are some differences from the various experiment done in this project, but most of the results stay within an acceptable range compared to the basic No Filter experiment. Both of the models performs differently, looking at the recall values, they are in general similar in correctly predicting real videos. Whereas Model Nr.1 is mostly better at predicting deepfakes, and is thus better at generalizing than Model Nr.2. The surprising difference is that Model Nr.2 is better at predicting and generalizing to blurred videos than Model Nr.1.

With regards to the second research question, which image filter or effects are the most influential in the detection of deepfakes. The results depends a bit on which model is used. Commonly for both models is that Gaussian Noise and Upside Down rotation results in much worse performance. With regards to Gaussian Noise, both of the models are bad at detection deepfakes, predicting them as real videos instead. The upside down rotation on the other hand is affecting the two models differently, where Model Nr.1 get a worse recall value on real videos, and Model Nr.2 is the opposite and becomes worse in regards to deepfakes.

Specific for Model Nr.1 is that 90 degree rotation and the appliance of blur have big affect on the results. Noticeably, the higher the blur level, the higher the influence is on the result. The 90 degree rotation makes Model Nr.1 bad at predicting real videos, whereas the appliance of blur lower the recall value for both deepfakes and real videos.

However with Model Nr.2, the appliance of blur is not affecting the model as much as Model Nr.1, but it affects it in a different way. The big difference is that applying blur results in a higher recall value for deepfakes, but worse on real videos. This has the opposite effect than it has on Model Nr.1.

Finally the third research question, which image filter or effects are the least influential in the detection of deepfakes. Just like with the other results, the answer is a bit dependent on the model used. On a general basis, black and white, lowered resolution, random rotation and mirrored videos are the least influential on both models. Among these, random rotation was the one with the biggest deviation from the standard no filter result. This was not surprising given the hypotheses. Additionally with regards to Model Nr.2 the 90 degree rotation performed similar to the No Filter experiment.

7.1 Further Work

Deepfake detection is still a field where there is still a lot of research and work left to be done. Reliable and good systems for automatic detection of deepfakes are still not in place. The results from this project will hopefully be a good contribution to further research, and be interesting to other researcher within the space.

Considering how this project can be taken further, it would be interesting to study and experiment with more filters and effects. It would be exciting to see how common image filters from major social media platforms as Instagram and TikTok would influence the results. In addition to using a more varied dataset, with deepfakes and real videos taken from several different sources. Which is something that most likely would end in results that are more applicable to real world scenarios.

Deepfakes are also constantly evolving and getting better. Meaning that as deepfakes improve, the detection models have to improve as well. This means that there is a constant need for improvements and research to close the gap between deepfake creation and detection.

Bibliography

[1] D. Fallis, 'The Epistemic Threat of Deepfakes,' *Philosophy & Technology*, vol. 34, no. 4, pp. 623–643, Dec. 2021, ISSN: 2210-5441. DOI: 10.1007/s13347-020-00419-2. [Online]. Available: https://doi.org/10.1007/s13347-020-00419-2.

[2] (). 'Tom (@deeptomcruise) TikTok | Watch Tom's Newest TikTok Videos.' en, [Online]. Available: https://www.tiktok.com/@deeptomcruise (visited on 28/04/2022).

[3] J. Damiani, 'A Voice Deepfake Was Used To Scam A CEO Out Of $243,000,' en, *Forbes*, 3rd Sep. 2019. [Online]. Available: https://www.forbes.com/sites/jessedamiani/2019/09/03/a-voice-deepfake-was-used-to-scam-a-ceo-out-of-243000/ (visited on 25/03/2019).

[4] C. Wang, 'Deepfakes, Revenge Porn, And The Impact On Women,' en, *Forbes*, 1st Nov. 2019. [Online]. Available: https://www.forbes.com/sites/chenxiwang/2019/11/01/deepfakes-revenge-porn-and-the-impact-on-women/ (visited on 25/03/2021).

[5] S. Burgess, 'Ukraine war: Deepfake video of Zelenskyy telling Ukrainians to 'lay down arms' debunked,' en, *Sky News*, 17th Mar. 2022. [Online]. Available: https://news.sky.com/story/ukraine-war-deepfake-video-of-zelenskyy-telling-ukrainians-to-lay-down-arms-debunked-12567789 (visited on 27/04/2022).

[6] J. Vincent, 'Facebook contest reveals deepfake detection is still an "unsolved problem",' en, *The Verge*, Jun. 2020. [Online]. Available: https://www.theverge.com/21289164/facebook-deepfake-detection-challenge-unsolved-problem-ai (visited on 12/04/2021).

[7] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies and M. Nießner, 'Faceforensics++: Learning to detect manipulated facial images,' 2019. DOI: 10.48550/ARXIV.1901.08971. [Online]. Available: https://arxiv.org/abs/1901.08971.

[8] B. Dolhansky, J. Bitton, B. Pflaum, J. Lu, R. Howes, M. Wang and C. Canton-Ferrer, 'The deepfake detection challenge dataset,' *CoRR*, vol. abs/2006.07397, 2020. arXiv: 2006.07397. [Online]. Available: https://arxiv.org/abs/2006.07397.

[9] L. Jiang, W. Wu, R. Li, C. Qian and C. C. Loy, 'Deeperforensics-1.0: A large-scale dataset for real-world face forgery detection,' *CoRR*, vol. abs/2001.03024, 2020. arXiv: `2001.03024`. [Online]. Available: `http://arxiv.org/abs/2001.03024`.

[10] J. P. Dasilva, K. M. Ayerdi and T. M. Galdospin, 'Deepfakes on twitter: Which actors control their spread?' *Media and Communication*, vol. 9, no. 1, pp. 301–312, 2021, ISSN: 2183-2439. DOI: `10.17645/mac.v9i1.3433`. [Online]. Available: `https://www.cogitatiopress.com/mediaandcommunication/article/view/3433`.

[11] T. T. Nguyen, Q. V. H. Nguyen, D. T. Nguyen, D. T. Nguyen, T. Huynh-The, S. Nahavandi, T. T. Nguyen, Q.-V. Pham and C. M. Nguyen, 'Deep learning for deepfakes creation and detection: A survey,' 2019. DOI: `10.48550/ARXIV.1909.11573`. [Online]. Available: `https://arxiv.org/abs/1909.11573`.

[12] L. Guarnera, O. Giudice, C. Nastasi and S. Battiato, 'Preliminary forensics analysis of DeepFake images,' Sep. 2020. DOI: `10.23919/aeit50178.2020.9241108`. [Online]. Available: `https://doi.org/10.23919%2Faeit50178.2020.9241108`.

[13] (). 'This Person Does Not Exist,' [Online]. Available: `https://thispersondoesnotexist.com/` (visited on 01/05/2022).

[14] I. Perov, D. Gao, N. Chervoniy, K. Liu, S. Marangonda, C. Umé, M. Dpfks, C. S. Facenheim, L. RP, J. Jiang, S. Zhang, P. Wu, B. Zhou and W. Zhang, 'Deepfacelab: Integrated, flexible and extensible face-swapping framework,' 2021. arXiv: `2005.05535v5 [cs.CV]`.

[15] Y. Nirkin, Y. Keller and T. Hassner, 'Fsgan: Subject agnostic face swapping and reenactment,' 2019. DOI: `10.48550/ARXIV.1908.05932`. [Online]. Available: `https://arxiv.org/abs/1908.05932`.

[16] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen and T. Aila, 'Alias-free generative adversarial networks,' 2021. DOI: `10.48550/ARXIV.2106.12423`. [Online]. Available: `https://arxiv.org/abs/2106.12423`.

[17] (). 'GitHub - NVlabs/stylegan3: Official PyTorch implementation of StyleGAN3,' [Online]. Available: `https://github.com/NVlabs/stylegan3` (visited on 23/05/2022).

[18] Faceswap. (). 'Faceswap,' [Online]. Available: `https://faceswap.dev` (visited on 04/03/2022).

[19] D. Web. (). 'Deepfakes web | make your own deepfake!' [Online]. Available: `https://deepfakesweb.com` (visited on 04/03/2022).

[20] Reface. (). 'Reface. Face swap videos,' [Online]. Available: `https://hey.reface.ai/` (visited on 01/05/2022).

[21] ZAO. (). 'Download zao app deepfake,' [Online]. Available: `https://zaodownload.com/download-zao-app-deepfake` (visited on 04/03/2022).

[22] M. Nuñez, ‘Snapchat and TikTok Embrace ‘Deepfake’ Video Technology Even As Facebook Shuns It,’ en, *Forbes*, 8th Jun. 2019. [Online]. Available: `https://www.forbes.com/sites/mnunez/2020/01/08/snapchat-and-tiktok-embrace-deepfake-video-technology-even-as-facebook-shuns-it/` (visited on 30/04/2022).

[23] WOMBO. (). ‘Wombo - ai powered lip sync app,’ [Online]. Available: `https://www.wombo.ai` (visited on 04/03/2022).

[24] Kaggle. (). ‘Deepfake detection challenge - identify videos with facial or voice manipulations,’ [Online]. Available: `https://www.kaggle.com/c/deepfake-detection-challenge` (visited on 05/04/2022).

[25] Kaggle. (2020). ‘Deepfake detection challenge - identify videos with facial or voice manipulations,’ [Online]. Available: `https://www.kaggle.com/competitions/deepfake-detection-challenge/data` (visited on 12/04/2022).

[26] Facebook-AI. (Jun. 2020). ‘Deepfake detection challenge dataset,’ [Online]. Available: `https://ai.facebook.com/datasets/dfdc/` (visited on 12/04/2022).

[27] J. Brownlee. (Apr. 2019). ‘How to Configure Image Data Augmentation in Keras,’ [Online]. Available: `https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/` (visited on 24/05/2022).

[28] B. Dolhansky, R. Howes, B. Pflaum, N. Baram and C. Canton-Ferrer, ‘The deepfake detection challenge (DFDC) preview dataset,’ *CoRR*, vol. abs/1910.08854, 2019. arXiv: `1910.08854`. [Online]. Available: `http://arxiv.org/abs/1910.08854` (visited on 12/04/2022).

[29] L. Jiang. (). ‘DeeperForensics-1.0/dataset at master · EndlessSora/DeeperForensics-1.0 · GitHub,’ [Online]. Available: `https://github.com/EndlessSora/DeeperForensics-1.0/tree/master/dataset` (visited on 19/04/2022).

[30] ondyari. (). ‘FaceForensics++: Learning to Detect Manipulated Facial Images,’ [Online]. Available: `https://github.com/ondyari/FaceForensics` (visited on 20/04/2022).

[31] C. C. Ferrer, B. Dolhansky, J. B. Ben Pflaum, J. Pan and J. Lu, ‘Deepfake detection challenge results: An open initiative to advance ai,’ nb, [Online]. Available: `https://ai.facebook.com/blog/deepfake-detection-challenge-results-an-open-initiative-to-advance-ai/` (visited on 10/04/2022).

[32] K. Zhang, Z. Zhang, Z. Li and Y. Qiao, ‘Joint face detection and alignment using multitask cascaded convolutional networks,’ *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016. DOI: `10.1109/lsp.2016.2603342`. [Online]. Available: `https://doi.org/10.1109%2Flsp.2016.2603342`.

[33] M. Tan and Q. V. Le, 'Efficientnet: Rethinking model scaling for convolutional neural networks,' 2019. DOI: 10.48550/ARXIV.1905.11946. [Online]. Available: https://arxiv.org/abs/1905.11946.

[34] S. Seferbekov, *Selimsef/dfdc_deepfake_challenge*, original-date: 2020-06-06T15:22:31Z. [Online]. Available: https://github.com/selimsef/dfdc_deepfake_challenge (visited on 05/04/2022).

[35] C. U. I. Hao, *Cuihaoleo/kaggle-dfdc*, original-date: 2020-04-28T23:05:28Z, May 2022. [Online]. Available: https://github.com/cuihaoleo/kaggle-dfdc/blob/91dae24a31caf6a3ca273e2b5d7337b9fe6f52d5/Model_Summary.pdf (visited on 25/05/2022).

[36] F. Chollet, 'Xception: Deep learning with depthwise separable convolutions,' 2016. DOI: 10.48550/ARXIV.1610.02357. [Online]. Available: https://arxiv.org/abs/1610.02357.

[37] T. Hu, H. Qi, Q. Huang and Y. Lu, 'See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification,' 2019. DOI: 10.48550/ARXIV.1901.09891. [Online]. Available: https://arxiv.org/abs/1901.09891.

[38] R. R. Salas, E. Dokladalova and P. Dokládal, 'Rotation invariant cnn using scattering transform for image classification,' 2021. DOI: 10.48550/ARXIV.2105.10175. [Online]. Available: https://arxiv.org/abs/2105.10175.

[39] C. U. I. Hao, *Cuihaoleo/kaggle-dfdc*, original-date: 2020-04-28T23:05:28Z. [Online]. Available: https://github.com/cuihaoleo/kaggle-dfdc (visited on 05/04/2022).

[40] E. Obrestad. (). 'Openstack at NTNU - SkyHigh - NTNU Wiki,' [Online]. Available: https://www.ntnu.no/wiki/display/skyhigh (visited on 26/04/2022).

[41] J. Brownlee. (Nov. 2016). 'What is a Confusion Matrix in Machine Learning.' en-US, [Online]. Available: https://machinelearningmastery.com/confusion-matrix-machine-learning/ (visited on 26/05/2022).

[42] A. Mishra. (Feb. 2018). 'Metrics to Evaluate your Machine Learning Algorithm.' en, [Online]. Available: https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234 (visited on 26/05/2022).

[43] K. P. Shung. (Mar. 2018). 'Accuracy, Precision, Recall or F1?' en, [Online]. Available: https://towardsdatascience.com/accuracy-precision-recall-or-f1-331fb37c5cb9 (visited on 14/05/2022).

[44] G. Dembla. (Nov. 2020). 'Intuition behind Log-loss Score.' en, [Online]. Available: https://towardsdatascience.com/intuition-behind-log-loss-score-4e0c9979680a (visited on 12/05/2022).

[45] (). 'Sklearn.metrics.log_loss.' en, [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.log_loss.html (visited on 10/05/2022).

[46] S. Narkhede, 'Understanding AUC - ROC Curve,' en, *Medium*, Jun. 2018. [Online]. Available: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5 (visited on 14/05/2022).

[47] (). 'Sklearn.metrics.roc_auc_score,' [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html (visited on 11/05/2022).

Appendix A

Additional Material

A.1 Confusion Matrix Results

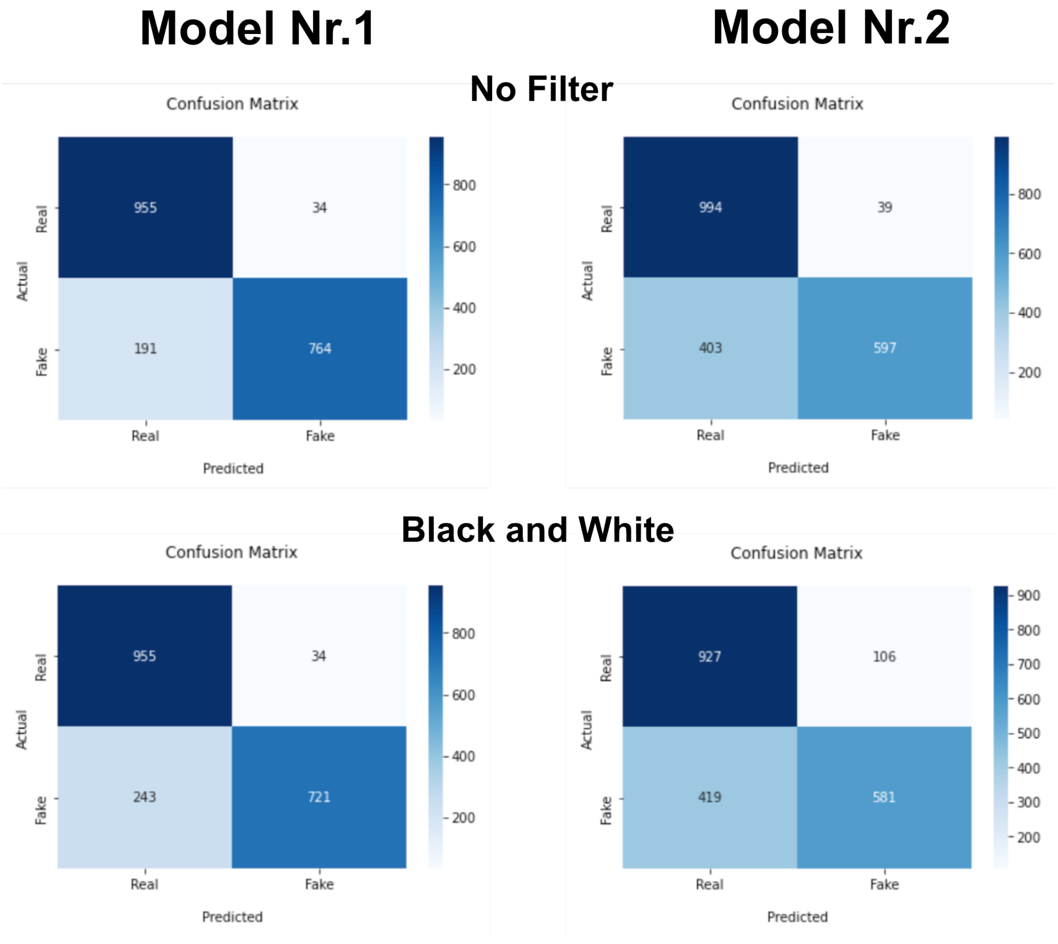


Figure A.1: Confusion Matrix Results - No Filter and Black and White

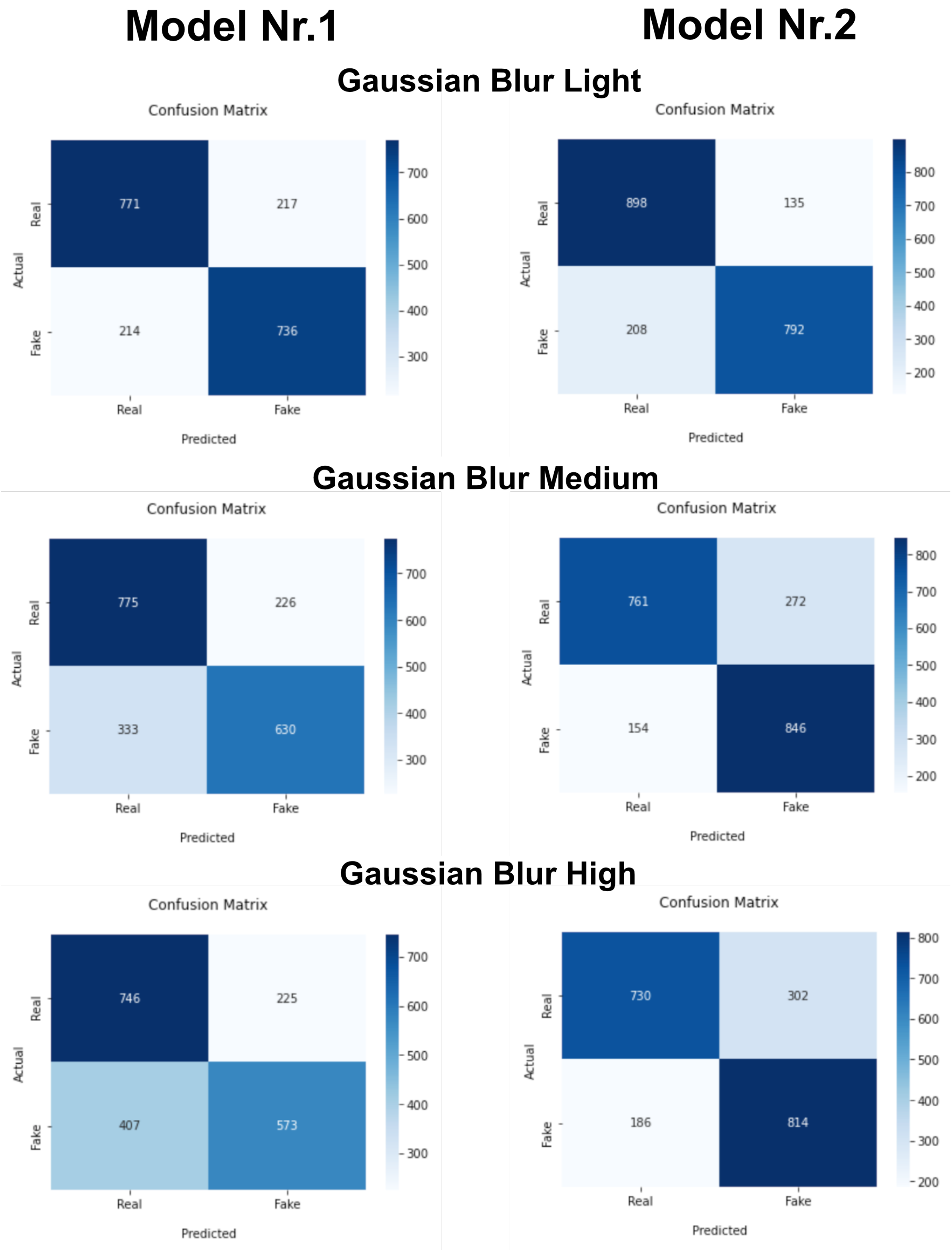


Figure A.2: Confusion Matrix Results - Gaussian Blur Light, Medium, and High

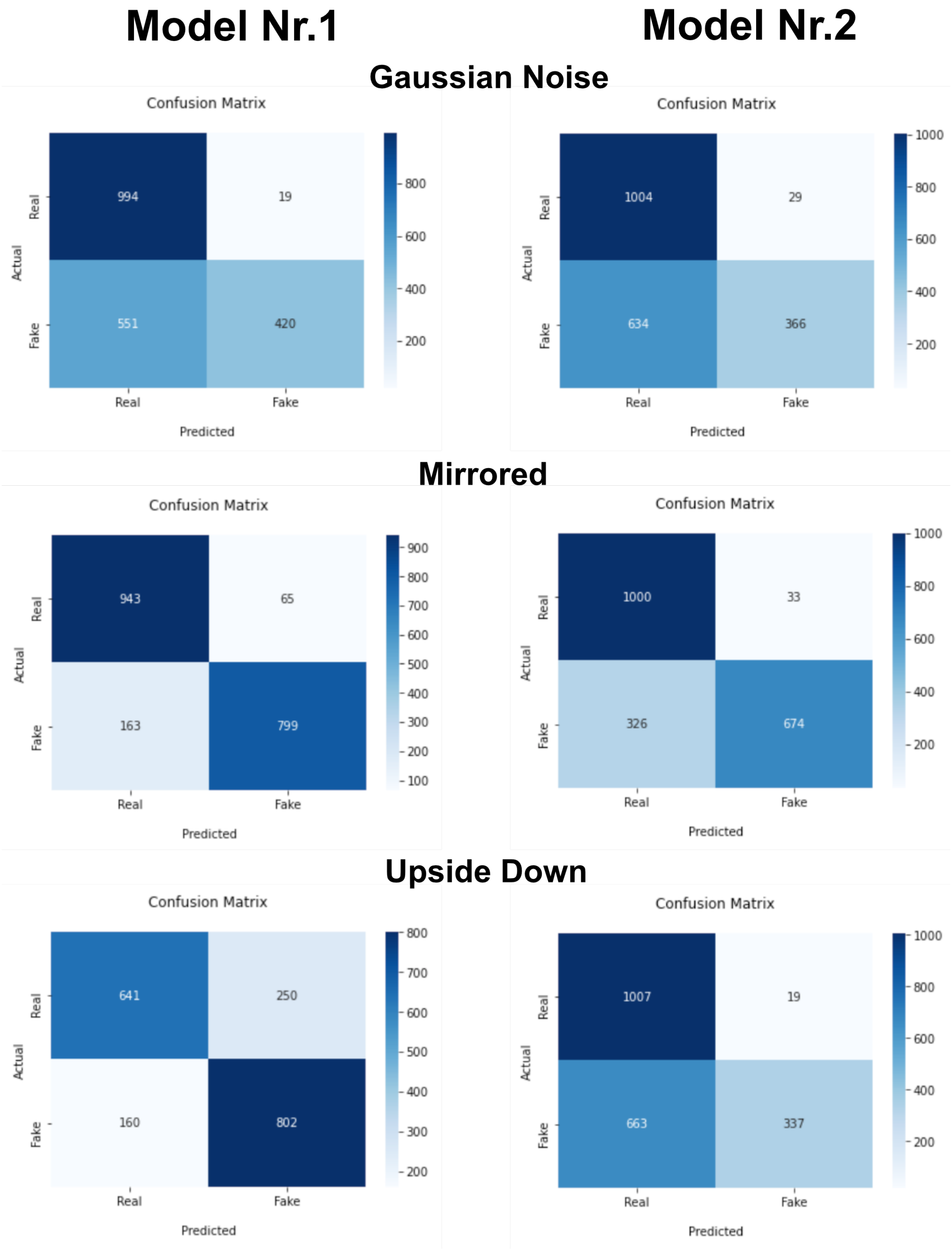


Figure A.3: Confusion Matrix Results - Gaussian Noise, Mirrored, and Upside Down

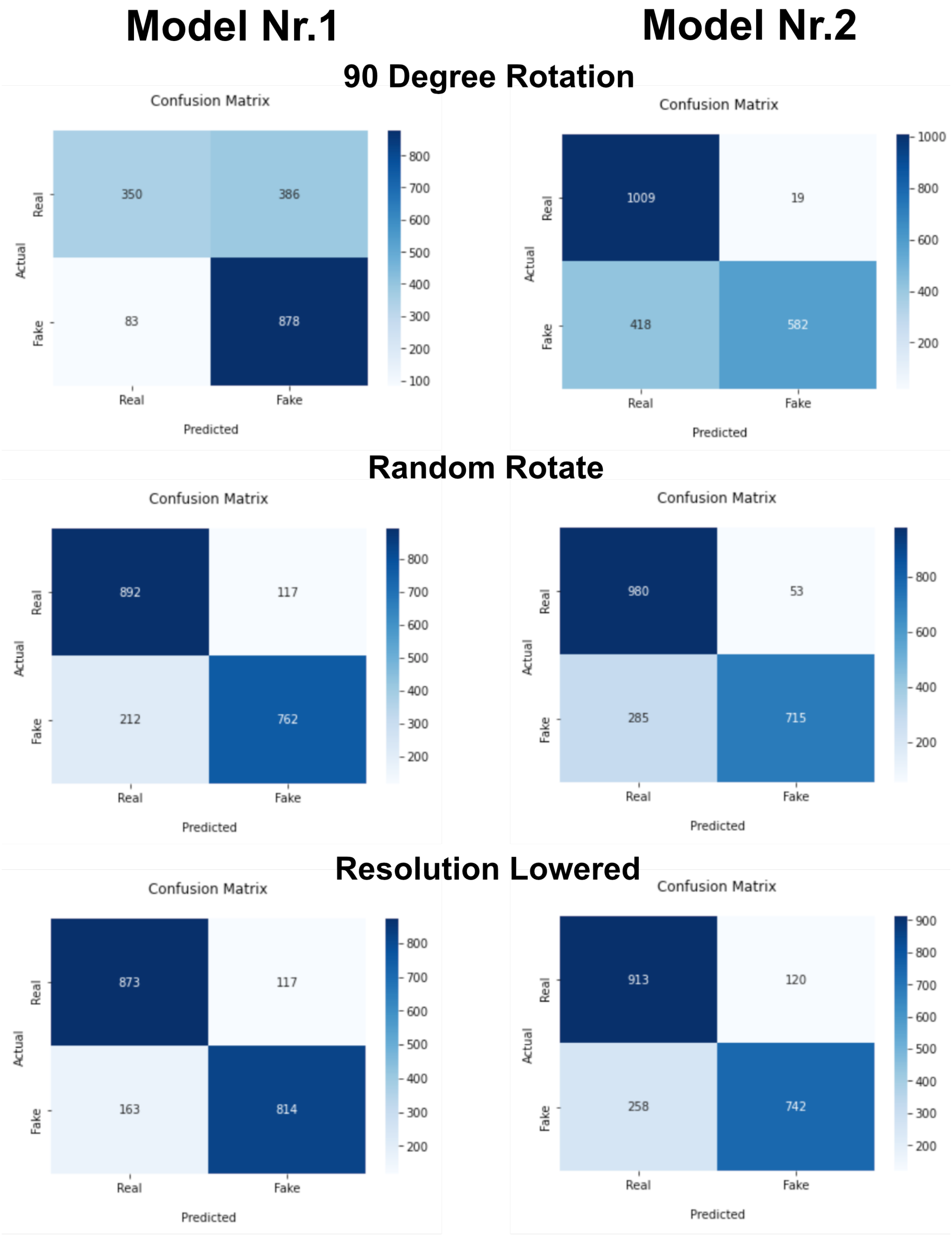


Figure A.4: Confusion Matrix Results - 90 Degree Rotation, Random Rotate and Resolution Lowered

A.2 ROC AUC Score

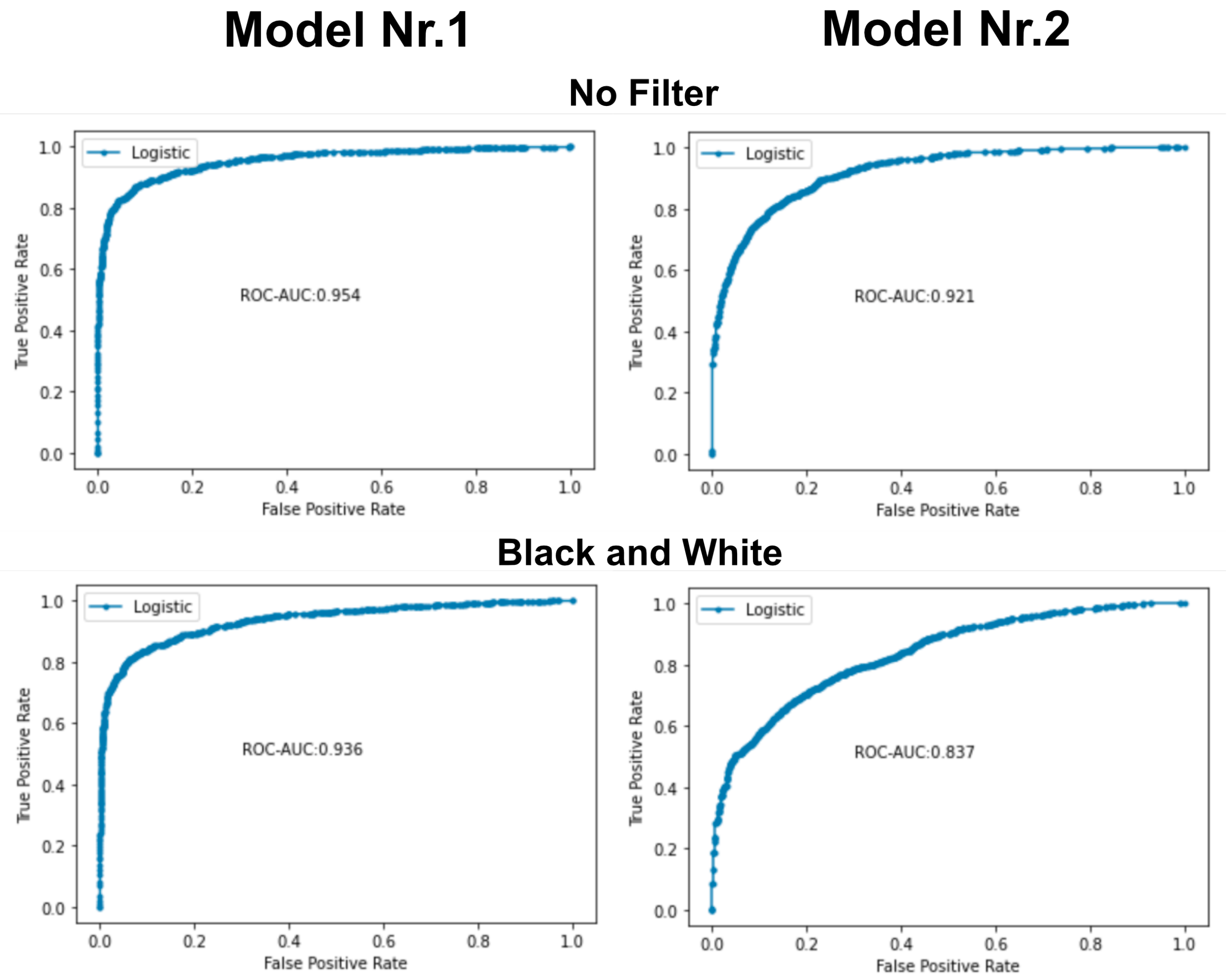


Figure A.5: AUC-ROC Results - No Filter and Black and White

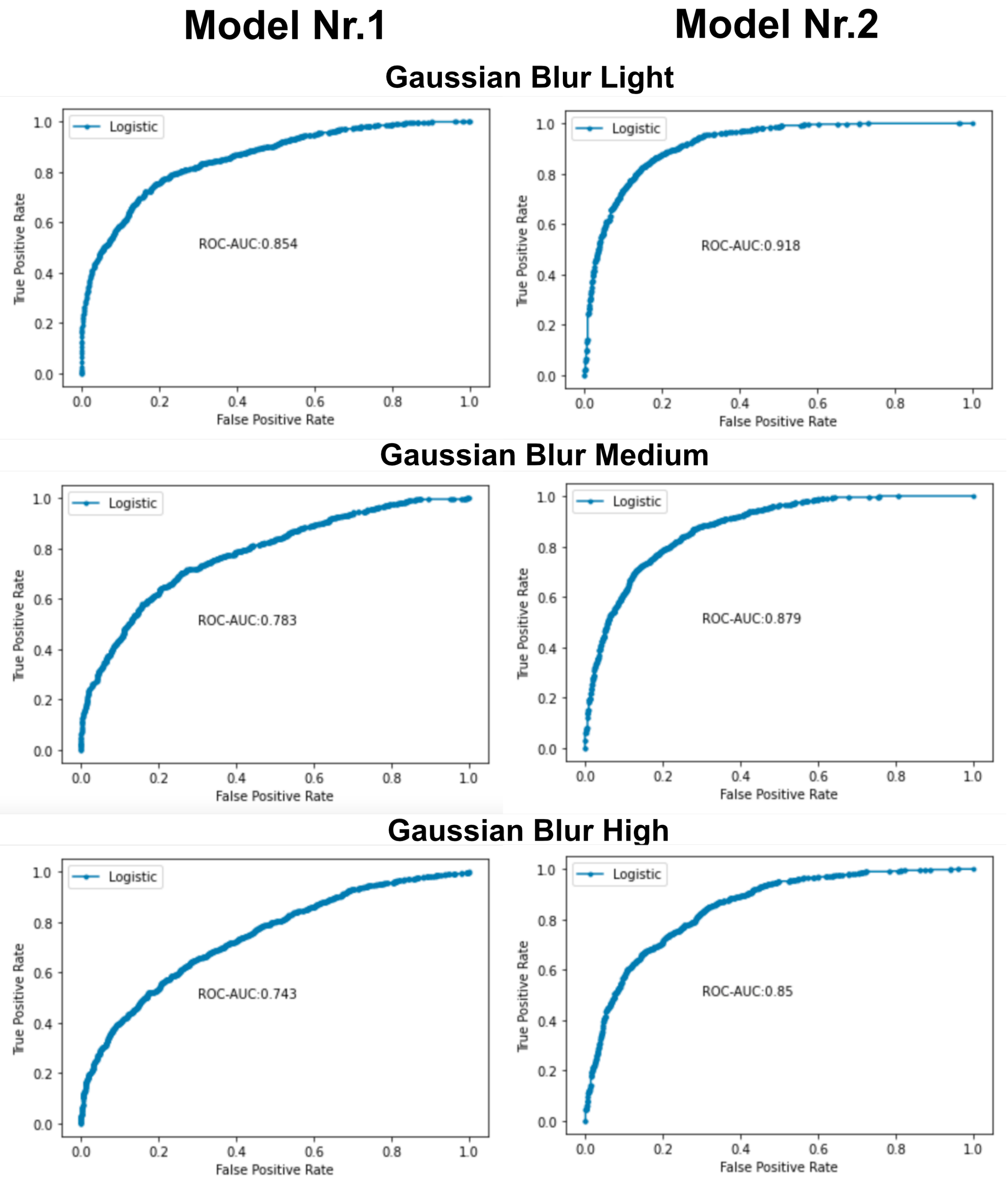


Figure A.6: AUC-ROC Results - Gaussian Blur Light, Medium and High

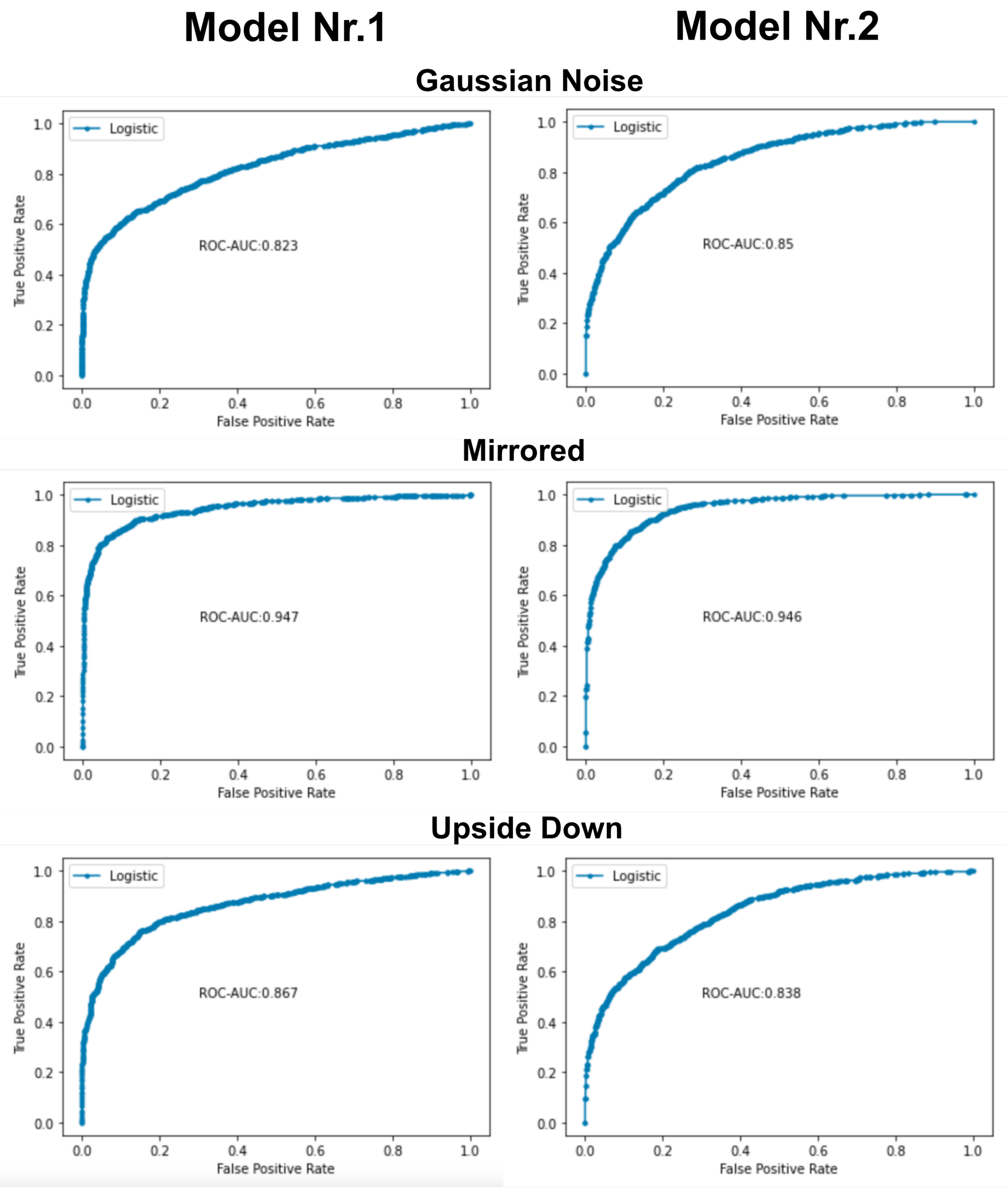


Figure A.7: AUC-ROC Results - Gaussian Noise, Mirrored and Upside Down

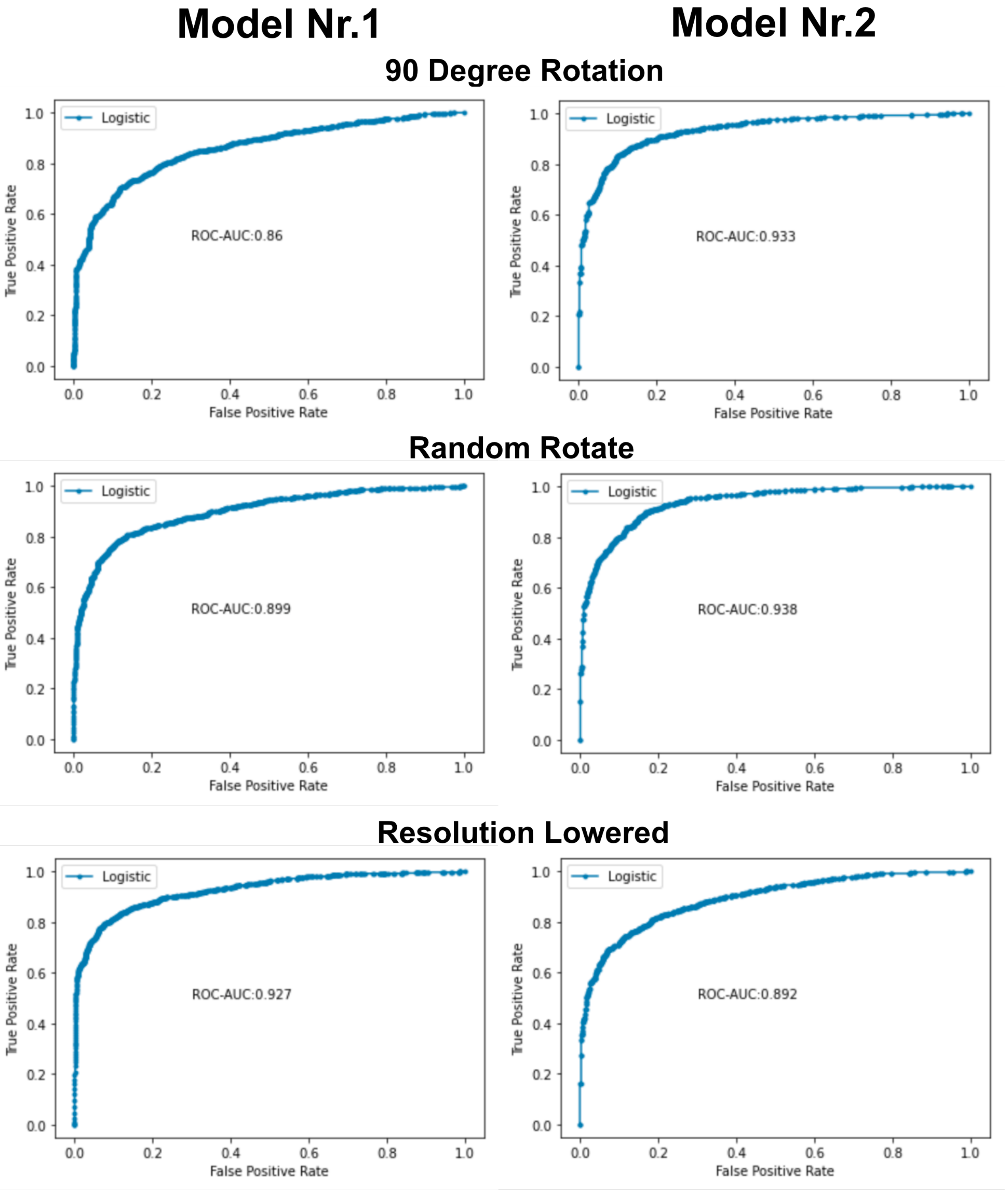


Figure A.8: AUC-ROC Results - 90 Degree Rotation, Random Rotate and Resolution Lowered

NTNU

Norwegian University of
Science and Technology