Northumbria University
NEWCASTLE

**Department of Computer & Information Sciences**

# Northumbria University
# At Newcastle
# 2024-2025

**B.Sc. Hons Computer Science**

**Name - Anukriti Sachan**

**Student ID - 21037581**

**Project Supervisor – Dr Naveed Anwar**

**Project Title - Deep Fake Detection on Social Media to Prevent Political Misinformation**

**Word Count - 11005 words**

**OneDrive link -** Dissertation

**Web Application Link – https://w21037581.nuwebspace.co.uk/home.html**

## Acknowledgement

Firstly, I would like to thank my supervisor, Dr Naveed Anwar, for helping me throughout the project; his constant guidance and feedback have been invaluable in shaping my research and practical project.

I would also like to thank my parents and my brother for being there for me throughout while giving me motivation and sacrifices to get me here to study abroad. I hope to make them proud through this achievement and fulfil their dreams, which they sacrificed to give me and my brother a great life.

To my grandmother, Nani, who sadly passed away during this project. I wish you were here to see the work I have accomplished. Thank you to my grandparents, whose belief in me has encouraged me to create a positive change in the world and show kindness.

## Abstract

Due to the launch of many generative artificial intelligence tools like Sora, DeepFaceLab (Petrov et al., 2021) and Dall-E 3 (OpenAI, 2024), it has gotten easier to create fake images of politicians doing or saying something which they never did using a small prompt, which pose a serious threat to democracy by eroding people's trust and integrity. The easiest targets are politicians, which can significantly mislead voters and spread misinformation. The public's trust has been shaken due to these images, where people always double-check every time, they read or see something online. This is why a robust deepfake detection tool tailored to compressed politicians' images was required. The present-day software and projects focus on using high-quality images, but social media often compresses the images and removes metadata from them when images are uploaded. This project aims to solve the problem of creating a politically based deepfake detection model which focuses on compressed images and make a platform which can both detect deepfakes and raise awareness by letting people understand what the main things are to look out for in an image to see if it is a deepfake. A custom dataset was curated by combining images from a variety of datasets, which included StyleGAN-StyleGAN2, Politics 101, and the Deepfake incident database. The dataset was curated keeping in mind to avoid demographic bias, so images of politicians from all around the world were added. The images were then pre-processed using Laplacian filtering, embossing, and FFT to display texture inconsistencies and frequency differences. A custom-made CNN model was trained from scratch and compared with a fine-tuned ResNet50 model. The model was created as a Tkinter GUI application which would in future work be included in the web application with the educational component. The model had the highest accuracy of 97.8% with a custom-built CNN when image processing techniques were applied. The project also had an educational component called "Did it really happen? (Political Version)", where users could upload an image, and the model would then classify the images as real or fake within 2 seconds. The web application included a quiz which helped users increase their skills in deepfake detection and gather cues to detect deepfakes easily. A qualitative analysis showed that users' scores increased by 131.58% after they were shown the hints to identify deepfakes. This indicated high educational value and an actual impact on users. This project helped contribute a unique way to detect deepfakes of politicians and choose the best image processing pipeline to help train the CNN-based models to predict deepfakes. Future work will focus on video deepfakes and applications for mobile devices to have it on the go.

**Keywords** – Political Deepfakes, deepfake detection tools, compressed political images, transfer learning

# Table of Contents

## List of Tables



## List of Figures



## List of Abbreviations –

1. CNN – Convolutional Neural Networks

2. AI - artificial intelligence
3. FFT – Fast Fourier Transform
4. SITD - Spitting Images: Tracking Deepfakes and Generative AI in Elections
5. PDID - Political Deepfakes Incidents Database

# Chapter 1

## Introduction

In 2022, a deepfake video of Ukraine's President Zelensky was circulating online asking Ukrainians to lay down their weapons. It was quickly discredited by forensic experts and removed (Burgess, 2022). Such content raises an urgent concern for democratic societies to help the public differentiate between fake and real content to prevent misinformation and distrust among voters. According to a survey done by Sippy et al. (2024) (Dennehy, 2024) for the Alan Turing Institute, out of 1403 people from the United Kingdom, 82.7% of people knew about deepfakes, and the most common targets were celebrities (50.2%) and politicians (34.1%). While people knew about deepfakes, 66.1% were unsure about their ability to detect a deepfake. 87.4% of people were concerned about its impact on elections. Current deepfake detectors only rely on high quality images and videos and fail when cropped images or videos from social media are uploaded, because social media platforms remove the metadata from the images and compress them highly (Dang-Nguyen et al. ,2023).

### *1.1 Motivation*

Early deepfakes were mainly used for making non-consensual pornography, mainly of actresses which led to ban of some subreddits on Reddit in 2018 (Hern, 2018). However, in recent years, there has been a rise of creating deepfakes of politicians to spread misinformation and try to manipulate public opinion. The quickest way to detect if a content is real or fake, is to look at its metadata because generative AI is still not able to replicate the metadata found on real content. Inadvertently, social media removes metadata and compresses the images to help their platform run smoothly with so much data, but this makes it extremely difficult to detect deepfakes quickly. The focus is only on politicians for this project because they are the easiest targets, and it directly affects the base of democracy by impacting public trust and stability of the government. The easy availability of politician's images is also a huge benefit to the people creating it as creation of a deepfake requires a lot of datasets for training to help create a realistic content (Auburn, 2024). By concentrating on politicians, this project will help increase knowledge amongst the public about deepfakes and help them easily detect deepfakes through the deepfake detector.

Deepfake is the content manipulated using artificial intelligence, which can now generate and manipulate images, audio, and video, and easily mimic real individuals. In recent years, deepfake technology uses generative adversarial networks (GANs) and convolutional neural networks (CNNs) to provide highly manipulated images (Islam et al., 2024). While this is extremely useful in industries like education and entertainment, its misuse in political

contexts—such as manipulating people's words or facial expressions—poses a high risk, threatening the integrity of democratic societies as it can reduce people's trust and create political misunderstandings (Jacobson, 2024). Because of how easy it is for a video to go viral, social media platforms such as Instagram, Facebook, and X often increase this risk by allowing manipulated images to reach a vast majority of audiences. Social media platforms often compress, resize, or crop images. These steps of processing images affect parts of the images that are key to detecting manipulations. Social media platforms often degrade the quality of videos and images and remove their metadata, which is an essential part of a forensic analysis of images, as various tools use to detect if the image is manipulated or not. This makes deepfakes, which are released during elections or protests, extremely difficult to trace and verify because of the high volume in which they are shared. Several tools, like DeepFaceLab (Petrov et al., 2021) and Dall-E 3 (OpenAI, 2024), are publicly available and extremely easy to use even by individuals without technical knowledge. The accessibility of such tools raises a concern about how easy it is to spread misinformation on a large scale. The literature review checked existing methods of deepfake detection. Then, it focused on how multiple layers of image processing techniques, combined with a machine learning model, addressed the challenges of low-quality data. It also discussed how this method identified and fixed the limitations of current approaches.

This project focuses on and addresses the following research questions:

- How effective are image filtering and machine learning in detecting deepfake images in political social media posts?
- Which image filtering techniques contribute most significantly in enhancing deep fake detection accuracy of politician images?
- Which techniques are the best for enhancing detection using image filtering?

### *1.2 Creation of Deepfakes*

Deepfakes are generated using a combination of neural networks and deep learning. Complex deep neural networks, including CNNs and GANs, manipulate images and videos. Such images and videos created are called deepfakes (Mubarak et al., 2023). People first started using such techniques for face swaps to manipulate videos and images. Furthermore, this technology has advanced so much that it can now manipulate facial expressions, speech and body movements. Deepfake technologies are advancing day by day, which makes it difficult to detect inconsistencies in images. Generative Adversarial Networks (GANs) consists of two mechanisms: a generator and a discriminator. A generator creates manipulated images, and a discriminator differentiates between real and fake images. The generator produces images that learn and mimic real-life photographs. Despite advancements in deepfake technology, subtle problems exist when creating manipulated images like symmetrical and smoothened-out faces. These features exist because of limitations in the training data and network architecture. The generator may overfit specific patterns, which leads to unnatural symmetry. Easy accessibility and rapid advancement of deepfake methods will have advantages and disadvantages. Political deepfakes are a substantial threat because they create misleading contexts. If this manipulated content goes viral on social media platforms, it severely impacts public opinion, which might affect elections (P and Sk, 2021). This presents an urgent

challenge for democratic societies to help distinguish between real and fake images earlier on, before it creates misinformation, which will improve public trust.

### *1.3 Practical Work –*

The project seamlessly integrates an interactive educational tool into its final deployment. The web-based tool is called **"Did it really happen? (Political Version)",** It is designed with two main functions. First, a user uploads an image and receives a classification of whether the image is real or fake. Next, a short quiz game is included, presenting users with real and fake images to identify. The users will first be shown some easy-to-identify signs in artificially generated images to help raise awareness. The main aim of this web application is to spread awareness and help build digital awareness amongst users, especially in politics. Combining a detection tool with a quick game combines a gamification tool into a deepfake detection tool. The program resizes images to 224x224 px, and then Laplacian filtering is applied with a kernel size of 3 and then embossing to highlight edges. Afterwards, the Fast Fourier Transform is used to compare frequencies. All images in the dataset were used from publicly available datasets to prevent privacy violations.

### *1.4 Context of the Project*

The project involves computer vision and an educational component. This project focuses mainly on compressed images of politicians.

After research using Google Trends for the keywords '**deepfake'** and '**deepfake detection'** worldwide during 2020-2025, it was visible that '**deepfake' peaked at 100** and **averaged at 34.4** while '**deepfake detection'** never **peaked more than seven** and **averaged 1.7** (Appendix C). This laid out a clear motive to create a project to detect compressed political deepfakes and increase awareness about deepfake detection tools.

This project is the first to address the challenge of detecting a deepfake in compressed political images, which are available on social media, using a combination of publicly available and custom-made datasets and applied multi-layered image processing techniques of Laplacian Filtering, embossing, Fast Fourier transform filtering and then compared with a custom CNN and transfer learning by using ResNet50 based on accuracy, precision, recall and F-1 score.

### *1.5 Aim of the Project -*

To develop and investigate an effective method to detect politicians' deepfake images using deep learning methods with image processing to prevent misinformation amongst the public.

***SMART Objectives –***

***Specific –***

- Completion of a comprehensive review of **≥ 30 papers** on deepfakes and detection techniques, comparing algorithms and finding limitations.
- Creating a custom-made dataset of **10,000 images**, including real and deepfake images from Politics 101, StyleGAN2.

***Measurable –***

- Image filtering techniques (Laplacian filtering, embossing, and Fourier transform) should be applied to the dataset to enhance and highlight its features and show deepfake differences like texture inconsistencies and uneven edges.
- Implementing machine learning models using filtered images to create a baseline model with accuracy ≥ **95**%.

***Achievable –***

- Development of a prototype application showing the system's detection accuracy and capabilities with a quiz to improve public detection scores.
- Evaluating the prototype through user testing of 10 participants.

***Relevant –***

- Training of machine learning models on filtered images to improve detection accuracy and comparing the performance of the baseline and transfer learning model ResNet50 using metrics.
- Analyse the feedback received from the participants and the analysis from the models to check the effectiveness of the techniques.

***Time-bound –***

- Complete literature review and finalise the 10,000-image dataset.
- Complete and submit Terms of Reference by week 6.
- Complete the final year project, including a report and the software

# Chapter 2

## Research and Planning

This project addresses the challenge of detecting compressed or cropped deepfakes of politicians on social media. This review helped critically analyse 35 peer-reviewed studies and articles which helped identify gaps and motivated this project's multi-layered image processing filters and comparison between custom-made CNN and ResNet50 models for compressed political images. Also, existing datasets used for deepfake detection were analysed and examined. Multiple image processing techniques were explored to handle compressed images. Then the research gaps helped to create the project plan.

### *2.1 Literature Review -*

#### *2.1.1 Background of Deepfakes –*

Bitouk et al., (2008) laid the foundation for facial manipulation by creating of the first deepfake through face swapping. They developed a system that automatically replaces faces in photographs in one second using C++. It had numerous limitations, such as less diversity and the same poses. Their main criteria was that the images needed to match poses and lighting so that they can easily blend into each other and when images were compressed, it was difficult to identify a face and led to the failure of the application. Goodfellow et al. (2014) developed **Generative Adversarial Networks** (GANs) which paved the way for making convincing deepfakes easily. Since then, the world has seen a rise in deepfakes because of open-source frameworks like **DeepFaceLab** (Petrov et al., 2021) and **FaceApp**. DeepFaceLab created high-quality face-swapping videos. FaceApp was developed in Russia and became extremely popular in 2018 (Eadicicco, 2019). This app allowed users to create deepfakes easily, especially with minimal technical knowledge. The app had demographic bias. (Morse, 2017; Varsha, 2023). Such open-source tools made it easier to create misinformation and pose ethical challenges in the political context.

Karras, Laine and Aila (2019) proposed a new generative architecture for GANs called Style-GANs. They curated a dataset of 70,000 images called Fliqr-Faces-HQ (FFHQ), which was much more diverse than the most used dataset, CelebA-HQ. This method created really convincing deepfakes, but Karras et al. (2020), after more research, discovered several issues in Style-GAN, including a droplet effect visible in most of the images around 64x64 resolution produced because of instance normalisation. To overcome this, they proposed a new method called StyleGan2. This method replaced normalization with demodulation to remove the artifacts available in the first version. This is relevant to understanding the creation of images and finding flaws within the creation process. Due to creation of this,

previous deepfake detection methods failed, until and unless they were retrained to include StyleGan2's dataset as well.

Online social media forums like Reddit's subreddit **"s/deepfakes"** gained massive popularity in 2017 and were later banned in February 2018 due to the creation of pornographic content of actresses (Hern, 2018). According to the research by Gamage et al. (2022) using Natural Language Processing on that subreddit, it was discovered that the main content which users searched for was related to pornography and political discussion. This research supported the idea that Reddit is often a place where people turn to view NSFW items. It was also verified through this research that users would often discuss the creation of deepfakes, which created a safe space for deepfake creators to improve the quality of their deepfakes. This unregulated community created an urgency for deepfake detection tools to protect citizens of democratic societies.

Tahir et al. (2021) identified the need for a tool that created awareness about deepfakes and helped the vulnerable understand the differences between a deepfake and a real image. They then designed a training program which made users understand the fundamental differences between deepfakes and real images and how to detect them. This study was a critical phase in the beginning of deepfake detection models. It gave insights about the different approaches machines and humans took to differentiate deepfakes from real ones (Gamage et al., 2022; Tahir et al., 2021). Even though this project was worthwhile, it did not allow users to check their knowledge, which our project's quiz will provide by displaying Grad-CAM visuals of every image displayed in the quiz. One of the first works of deepfake detection was using a CNN by Guarnera, Giudice, and Battiato (2020). They focused on detection using forensic traces hidden inside images using Expectation Maximisation. This study helped further research understand the use of neural networks in deepfake detection techniques. They achieved 99.31% accuracy on StyleGAN vs StyleGAN2 datasets in their project, but did not mention any testing on in-the-wild datasets. Sudarshana and Vamsidhar (2025) discovered that this method often failed due to overfitting in particular regions of the datasets it is trained on.

In summary, deepfake technology has raised serious concerns about spreading political misinformation. Politicians are the easiest targets due to the huge impact their deepfakes could case. This helped with understanding that even though deepfake technology has advanced, there will always be minor artifacts which will help people identify fake from real images. Therefore, this project proposes a hybrid approach of detecting faces in images, preprocessing images using filtering and then feeding them to a custom-made CNN and ResNet50 to detect hidden artefacts in compressed political images and a need for a application which provides the public knowledge about deepfake and their detection.

### *2.1.2 Impact of Deepfakes on Political Posts on Social Media (Case Studies)*

Social media has seen an extreme rise in deepfakes, as discussed in 2.1. There is a massive need for deepfake detection technology for images found on social media. There have been various instances where deepfakes have created misinformation and distrust in the public. Such real-life examples are –

Dhanuraj and Solomon (2024) discuss in their paper about Generative AI and its influence on **Indian elections** about an incident when deepfake videos of A-list Bollywood celebrities like Amir Khan, Amitabh Bhacchan and Ranveer Singh, were created where they mentioned about the promises of Prime Minister of India Modi unfulfilled in his previous term and the video ended with "**Vote for Justice, Vote for Congress**". This video got 0.5 million views in a short span of time. Such videos harm the opposition by tarnishing their image and swaying people's votes and ideologies. This deepfake highlighted the ineffectiveness of current detectors on social media platforms, as they are mostly trained on Western datasets.

Taylor Swift had to cancel her concert in Vienna in August 2024 because of security threats from ISIS (Bell and Cooney, 2024). Current US President Donald Trump used deepfake images of Taylor Swift and her fans wearing "**Swifties for Trump**" t-shirts. The post also had a news article generated using AI, which said, "**Swifties turning to Trump after ISIS foiled Taylor Swift Concert**" (Robins-Early, 2024). Such tweets and images, when shared by President Donald Trump, added credibility to the images and had no warning of being deepfakes. This created a lot of misinformation and distrust among the public.

Bond (2024) also discusses when, in January, Democrats received a call from ex-US President Joe Biden asking them not to vote in the November New Hampshire primary elections. This call was later identified as an audio created using artificial intelligence by a Democratic consultant who said he did it to raise awareness about deepfakes. It clearly shows how easy it is to sway voters and create misinformation through deepfakes, even in an audio context.

According to Sebastian (2024), when the BBC investigated the use of deepfakes in Indian elections, they discovered from Divyendra Singh Jadoun, founder of The Indian Deepfaker (TID), that he had received multiple requests from politicians to create compromising images and morph videos and audios of their opposition party members to tarnish their image. This raises a significant risk throughout the entire country. India, being the largest democracy in the world, needs some good detection tools to identify deepfakes and reduce misinformation.

These days, deepfakes are extremely popular and easy to make for people with no technical knowledge. Generative AI models like Grok are unregulated, which makes it extremely difficult to reduce the production of such videos and images. According to Dave (2025), Grok has little to no oversight. It creates misleading information. It is also challenging to retract misinformation once it goes viral, creating ethical issues. Grok gets its information from X, which is a platform that contains innumerable stories and opinions that are entirely or half false. Grok learns from this and then gives responses to users with significantly fewer regulations.

These case studies help realise and confirm the relevance of focusing on just politicians. Images uploaded on social media get compressed, making it difficult for current technologies to identify them. This project will make a huge impact on a better detection tool and help public detect deepfakes as well.

### 2.1.3 Existing Deepfake Datasets

The next part for this project was selecting the data which will be split into train-test-val. Many datasets were researched, and their limitations were discussed to make a custom dataset.

**Politics 101** (Antoniou, 2020) - It is a dataset consisting of 10 politicians from all around the world. It has been used to help with real images. Nevertheless, due to a lack of diversity, more datasets have been added to it. However, this dataset will be extremely helpful in identifying images based on political backgrounds.

**FaceForensics**- (ondyari, 2022) This dataset is created using 1004 videos from YouTube. The resolution chosen for the dataset is greater than 480p. It has over half a million images (Rössler et al., 2018). Even though the dataset consists of many images, it does not focus on politicians. It has many news caster videos. The dataset also does not guarantee much diversity.

**Detect AI-Generated Faces**: **High-Quality Dataset**- This dataset contains 1001 artificially generated and 2202 real images (Rehman, 2024). As its owner describes it, it is ideal for deepfake detection. Even though it does not focus on politicians, it has recent images produced by generative AI to help our model identify new images and trends.

**StyleGan-StyleGan2 Deepfake Face Images**- This dataset comprises 7000 artificially made and 5890 real-life images (Bhargava, 2023). It is incredibly diverse and consists of various backgrounds. This dataset, mixed with more images, will be the perfect dataset for this project for training both real and fake images. Due to demoghraphic constraints, more datasets have been added to the custom dataset as well.

**Political Deepfakes Incidents Database** - Walker, Schiff and Schiff (2024) developed a dataset which includes all the incidents of deepfakes since 2017. They have collected data from comments and the captions surrounding social media posts related to deepfakes. This dataset has been used for training, testing and validation of fake images, but this cannot be used alone as it consists of only deepfake incidents. This dataset also lacks diversity because it only comprises images of Western politicians, but due to political context, images have been used in the dataset.

**Spitting Images: Tracking Deepfakes and Generative AI in Elections (SITD) -**Gorman et al. (2024) created a tool that shows the use of deepfake campaigns and generative AI during elections worldwide. This dataset again has a lot of deepfakes, but lacks a good demographic difference, because it is mostly focused on Western news.

**Custom Dataset** – Seeing how no existing dataset fits this project's aims completely, a custom dataset was curated from multiple sources comprising of 5000 real and 5000 fake images. StyleGan-StyleGan2 was used for fake images dataset to cover the current artifacts which GANs created. Multiple sources were used to collect images of real politicians from public domains to prevent privacy violations. The hardest part was to collect fake images of politicians because the only dataset available is the Political Deepfakes Incident Database but it is highly focused on western politicians. Images available online were added. During training the model, the compression percentage was added to curate to the project's aim.

| Dataset | Link | Images/Videos | No of Fakes | No of real | Political Context | Compressed? | Limitation |
|---|---|---|---|---|---|---|---|
| Politics 101 | (Antoniou, 2020) | Videos | 5639 | 590 | No | No | 10 world politicians only |
| FaceForensics++ | (ondyari, 2022) | Images and Videos | 1000 | 1000 | No | No | Lack of politics context |
| Detect AI-Generated Faces: High-Quality Dataset | Link | Images | 1001 | 2202 | No | No | Demographic bias |
| StyleGan-StyleGan2 Deepfake Face Images | Link | Images | 7000 | 5890 | No | No | No political context |
| 140k Real and Fake Faces | Link | Images | 70,000 | 70,000 | No | No | No political context |
| Human Faces Dataset | Link | Images | 4630 | 5000 | No | No | Demographic bias |
| Political Deepfakes Incidents Database | Link | Images and Videos | 1148 | 0 | Yes | Yes | No real images for comparison |
| Custom-Dataset | Available in OneDrive | Images | 5000 | 5000 | Yes | Yes | NA |

*Table 1 Datasets available for deepfake detection and its comparison with the custom made dataset*

To reduce demographic bias, multiple datasets from all around the world were used –

*Table 2 List of datasets used to reduce demographics*

| Continent | Country | Images | Reference | | | |
|---|---|---|---|---|---|---|
| **Asia** | India | 300 | (singh, 2021) | (Sahu, 2022) | | |
| | Russia | 47 | (Antoniou, 2020) | | | |
| | Pakistan | 47 | (SaadAli5, 2024) | | | |
| | Korean | 300 | (Song, 2023) | | | |
| | China | 10 | (Zeinabartaill, 2021) | | | |
| | Japan | 49 | (Antoniou, 2020) | | | |
| **Africa** | All | 510 | (Ajewole et al., 2024) | | | |
| **America** | All | 350 | (Ballew II and Todorov, 2007) | (Olivola and Todorov, 2010) | (Olivola et al., 2012) | (Todorov et al., 2005) |
| **Europe** | Italy | 73 | (Antoniou, 2020) | | | |
| | UK | 50 | (Antoniou, 2020) | | | |
| | Germany | 46 | (Antoniou, 2020) | | | |
| | Spain | 42 | (Antoniou, 2020) | | | |
| | France | 34 | (Antoniou, 2020) | | | |

| | | | | | |
|---|---|---|---|---|---|
| | Greece | 53 | (Antoniou, 2020) | | |
| **Australia** | Australia | 80 | (Antoniou, 2020) | | |

After researching on current datasets, a combined dataset was created which had focus on diversity and politicians. A significant time and effort were spent to curate the perfect dataset because the dataset plays a huge role in helping the model perform better in real life scenarios.

### 2.1.4 Existing Deepfake Detection Technologies

Deepfake detection being an important topic has a lot of existing detection technologies which use different methods to identify manipulated images. Some of those techniques were researched and their limitations are understood, and it was concluded which parts of those projects were helpful and then they were applied in this project.

A **deepfake detection using simple CNN** was created by Guarnera, Giudice, and Battiato (2020) – This was trained by cross training on multiple datasets and the highest accuracy was with CELEBA vs STYLEGAN, where the accuracy was 99.65%. It was trained using Expectation-Maximization algorithm which extracted some features from an image and then experimented with it. However, this approach failed on different datasets which concluded that the dataset struggled with different aspects like lighting, quality. Hence, it can be concluded that even though CNN are powerful, they do need some more algorithm with it.

**Deepfake detection of face images based on a CNN** - Kroiß and Reschke (2025) developed a system using transfer learning by using a pre-trained model called ResNet-50. They also pre-processed the dataset and sorted out images due to bad quality or the lack of a face present. They used the **train-validation-test** split approach to train their model and check its accuracy. They illustrated that training all layers in the first training step gave best results regarding transfer learning which was validated by Yosinski et al., (2014). Their approach had a precision of 0.98 and an accuracy of 96%. Their approach of **utilising a pre-trained model** and **fine-tuning pre-trained layers** helped improve the model's performance, instead of keeping fewer layers. It was understood that a custom CNN would be much easier to include the filtering in images, whereas ResNet50 comprises of a much better feature extraction techniques as it is a well-trained ImageNet model.

**FakeCatcher (Intel, 2022)** – A real-time deepfake detection system that analyses blood flow variations in videos developed by Intel (Ciftci and Demir, 2020). Their approach was that there are blood flow variations in real portrait videos, which are not yet available in fake videos. They have used the approach of **cross-training multiple datasets** to help compare which dataset works the best. Despite their innovative approach, this model is not entirely valid for the compressed videos on social media platforms. They have used CNN architecture to train their model. They claimed that their model had an **accuracy of 91.07%** in classifying in-the-wild videos. However, when James Clayton from BBC (Clayton, 2023) ran some tests on this system, it was discovered that the **quality of videos** was a high requirement to predict if a video is fake or not because the lower the quality, the harder it is to analyse or even detect blood flow variations, due to which it detected several in the wild videos found on social media which were real, to be fake and vice-versa. Their reliance on blood flow means they

need extremely high-quality videos. In contrast, social media platforms compress videos to help run the platforms smoothly, and this is the gap that this project is trying to fill by using images instead of videos to avoid reliance on quality, motion blur, and poor lighting, all of which can be fixed using image processing. This makes it suitable for social media platforms, where screenshots of videos can also be detected easily for quick use.

**Deepfakes Detection by Iris Analysis (Tchaptchet et al., 2025) –** A new technique was introduced by Tchaptchet et al. (2025) where they detected **characteristics of eye irises** by extracting pupils from faces and discovered that an image is real if the pupil is shaped elliptical or round, and if both the eyes have similar gradient maps of iris. If not, then the image is fake. The system received a **precision of 0.960** and **an accuracy of 0.979** on images produced by StyleGAN2. However, they have also identified limitations, such as the tool failing when the image **does not have proper lighting** and is **not of high quality**. The tool also might fail if the pupil shape is **irregular** because of some diseases (Guo et al., 2022). This research identified that **image processing** tells a lot about images that are not visible easily on the first try. They used the **Sobel operator** to compute horizontal and vertical gradients, which proved to be an extremely important process for efficiently extracting the features of the face. Unlike the approach of Tchaptchet et. al. (2025), which relies only on high-quality images, our proposed approach addresses the challenge of compressed images typically available on social media.

**Fake Face Detection tool** from TrustWorthy BiometraVision Lab, IISER Bhopal - (Lab, 2024, Agarwal and Ratha, 2024). It is a tool for which users do not need to install the code through GitHub to view it and is openly available on huggingface. After feeding an image of President Donald Trump (Sharma, 2023) which had been discredited as fake, the detector still detected it as a real image.

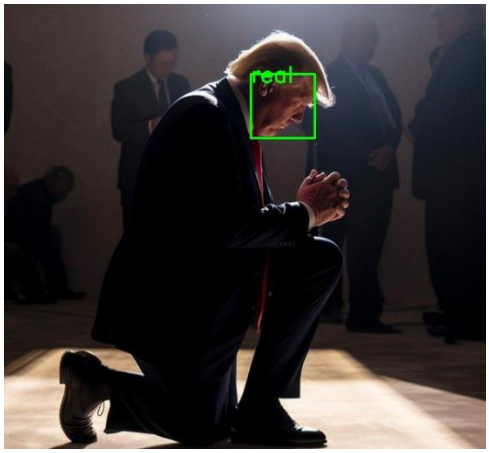


*Figure 1 AI generated image of President Donald Trump (Sharma, 2023) detected as real by a deepfake detector.*

| ***Method*** | ***Input*** | ***Quality of Content*** | ***Political Context*** | ***Accuracy (on the Dataset)*** | ***Testing on in-the-wild datasets*** | ***Limitations*** |
|---|---|---|---|---|---|---|
| **FakeCatcher (Intel, 2022)** | Video | High | No | 91.07% | Yes (Failed when testing | Failed on compressed videos |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | done on in-the-wild videos) | |
| **Guarnera, Giudice, and Battiato (2020)** | Images | High | No | 99.31% | No | overfits certain features. |
| **Iris Analysis (Tchaptchet et al., 2025)** | Images | High | No | 97.9% | No | Reliability on high-quality images |
| **ResNet50 (2025)** | Images | High | No | 96%. | No | No compressed images |
| **Proposed Project** | Images | Low | Yes | 97.8% | Yes | - |

*Table 3 Comparison of various components of Previous Works with the Proposed project*

According to the research done by Guarnera et al. (2020), it was concluded that the process of using Laplaican filtering revealed patterns in frequencies which are not present in real images which was revealed after applying Fourier transform. This project uses the same process with the addition of embossing of images because it is not creating new content but just emphasizing the features of images. Even though these detection methods offer unique analysis methods, such as real-time analysis in FakeCatcher or iris analysis, they all share common challenges. Almost all of them rely on high-quality videos or images and struggle to operate when the images or videos are compressed or cropped to show only some features of the face, which makes them less effective on content used on social media. That is why a hybrid approach combined with Laplacian, embossing and FFT, with a comparison between ResNet50 and CNN, was chosen, as supported by Kroiß and Reschke (2025).

### *2.1.6 Limitations of Datasets Used in Existing Methodologies*

Despite the tremendous progress in deepfake detection techniques, there are still several limitations that continue to cause problems with the real-world applicability of these systems.

**Biased images** raise a critical concern for the training datasets, especially when the dataset is only faces of celebrities, resulting in the detector performing poorly on less-seen groups (Mehrabi et al., 2021). This causes a deepfake detection model to generalise across various demographic groups, reducing effectiveness. The datasets found online often only include images from Western media, making it more difficult for the model to work when images are from a different geographical group, as discussed in section 2.1.2.

**Image quality** is extremely critical. Deep learning models are highly sensitive to the quality of input images (Ding and Li, 2018). Low-quality images on social media have been compressed and cropped. That is why I took the approach of enhancement techniques to help us understand these features much better.

The rapid evolution of deepfake creators is another challenge. As GANs and CNNs further improve, as discussed in Chapter 2.1.1, the quality of deepfakes also increases, which makes it difficult for existing methods to adapt well to the changes. The dataset must be

continuously extended and include recently manipulated content. However, maintaining such a huge dataset involves computational resources and continuous research.

### *2.1.5 Ways to Overcome Low-Quality Images*

Image processing techniques enhance edges and improve image quality to overcome the challenges of detecting deepfakes in low-quality or cropped images from social media. This project will use Laplacian filtering, embossing and the Fast Fourier Transform. Gaussian filters were not used because they reduce image noise, removing important information from images, which is very important for deepfake detection.

***Laplacian filtering*** **(Appendix B, Fig. 2.1)** – According to Paris et al. (2010), when Laplacian filters are applied to low-resolution images, they emphasise edges in an image. Deepfake generators often produce images with overly smooth or sharp edges; however, in real images, edges are often there due to natural lighting and uneven textures of faces. By applying the Laplacian filtering method, these differences become more detailed and will, therefore, allow the detection model to identify unnatural images or abrupt changes.
***Embossing*** **(Appendix B, Fig. 2.2)** –Embossing adds a 3d quality to images which refines the image and enhances the image processing algorithm (Pál et al., 2020). It adds a three-dimensional effect to images by contrasting between colours of images. It highlights details in images like wrinkles, hair strands, and skin texture. Since deepfake images lack such fine details, it becomes quite a useful technique to differentiate between genuine images and fake ones. ***Fast Fourier Transform*** **(Appendix B, Fig. 2.3)** – GANs often create extremely smooth images; therefore, they lack high frequencies available in real images. According to Ke and Wang, (2023) who created DF-UDetector, they analysed that even though their method of using FFT had an accuracy of 90.26%, it failed when it performed on compressed images, because of which this project focuses on multi-layered image processing techniques with the integration of FFT in the end. Addition of 1 confirms that log will never be 0. This helps distinguish between frequency domain peaks. The combination of all techniques creates a multi-layered effect on images, which exposes edges and contrasts. This is a robust approach to focus on fine details, which most tools usually miss out on, to differentiate actual images from fake ones. It would then boost detection accuracy as well.

Data augmentation is required during training for detecting compressed social media images. This helps the model learn different images with random compression, crop, and sizing. In summary, to overcome low-quality images, it is required to highlight the features which can overcome compression and train the images on the Fourier Transformed versions to help increase accuracy.

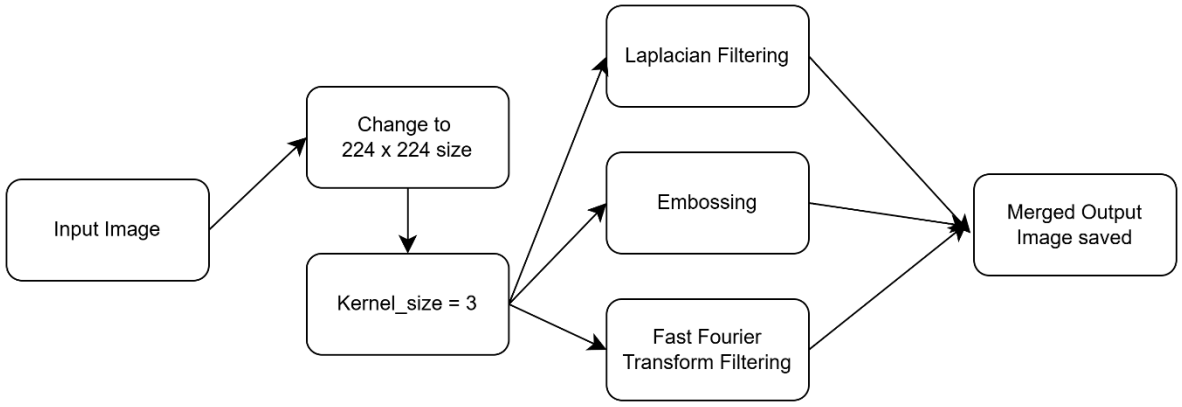


*Figure 2 Image Processing Pipeline*

### 2.1.6 Research Gaps

No existing work yet addresses deepfake detection for compressed images of politicians found on social media using a multi-layer image processing technique. Even though the public knows about deepfakes, their knowledge about deepfake detection techniques and tools is very low, which creates an urgent need for a detection model that can detect compressed images and has educational content to raise awareness. While FakeCatcher by Intel does have a groundbreaking approach, its model fails when it comes to in-the-wild videos, which have been compressed, and blood flow is not easily detectable. That is where this project will fill the gap and create a tool that will detect compressed images through image processing tools. After understanding the gaps through image processing and a machine learning architecture will increase the detection accuracy, which will work perfectly towards compressed social media images. This project will help the public by reducing political misinformation.

## 2.2 Planning

A 24-week project plan was constructed to ensure each objective was met on time (see Appendix A). No existing work uses multi-layered filters and CNN to detect deepfakes. After understanding the limitations of each study, a robust approach was taken, which included –

i. Custom-curated dataset to remove bias and help the model perform better in in-the-wild images. Include specific politicians' images because most of the politicians' images often include the same background and nice lighting.
ii. Pre-processing images using filters like Laplacian, embossing and FFT to reveal edges, which can be compressed on political images available on social media, and reveal frequency domains.
iii. Training the dataset on a custom CNN and fine-tuned ResNet50 to help with accuracy.
iv. Creation of a web application which includes both the detector and a quiz to help raise awareness amongst users and help them identify deepfakes manually.

Based on our research questions and literature gaps, seven "Must", five "Should/Could" and two "won't" functional and non-functional requirements were identified (MoSCoW table).

### 2.3 Summary

Nguyen et al. (2019) explain how the capability for generating interesting fake images and videos is improving and increasing at an alarming rate. The ongoing development of a more robust detection algorithm remains crucial to reducing its threats to society's privacy, security, and trust. By adding advanced image processing techniques with a machine learning model, this project will overcome existing limitations, enhancing integrity in social media content, reducing misinformation, and supporting fact checks in political contexts.

Continuous advancement in this area is significant for protecting people's trust and ensuring the integrity of social media posts. By using **FakeCatcher's (Intel, 2022)** approach of training the model on various datasets to see which works best, the one that will be used is the one that works best. **Deepfakes Detection by Iris Analysis (Tchaptchet et al., 2025)** uses the Sobel operator, which will be used in the multi-layered image processing techniques, as it has proved valuable. **Deepfake detection of face images based on a CNN - Kroiß and Reschke (2025)** used the approach of using a pre-trained model **ResNet 50** and fine tuning its layers to help improve the model, which is an approach which will be used in this project as well to see if my own created model works better or the pre-trained with fine tuning and adding a layer of my own.

In this chapter, various pieces of literature which gave insights about deepfakes were discussed alongside their creation, and how to tackle them. Numerous pieces of literature were also discussed, which helped understand what research others have done in this area and what solutions have been found. This review identified a significant gap in the current literature: the absence of an image-based detection tool tailored explicitly to compressed political images commonly found on social media.

# Chapter 3

## Proposed Practical Work

This chapter will discuss the framework for detecting deepfake images in the political context. Firstly, images will be pre-processed using OpenCV and then the best technique was compared between a convolutional neural network (CNN) and a pre-trained model called ResNet50 to enhance detection accuracy in compressed social media images. This methodology will discuss dataset creation, image processing, model training and performance evaluation.

### 3.1 – Design –

#### 3.1.1 Functional and Non-Functional Requirements -

| ***ID*** | ***Name*** | ***Descriptions*** | ***Source*** | ***Related Actor*** | ***Priority*** |
|---|---|---|---|---|---|
| F01 | Uploading Image | The system shall allow the user to upload an image for classification | Main objective | User | Must |
| F02 | Deepfake Classification | The system shall classify the image as 'Real' or Fake' and return the result | Main objective | User | Must |
| F03 | Grad-CAM Explanation | The system shall display a GRAD-CAM heatmap to display the regions which influenced the classification | User Testing | User | Won't |
| F04 | Quiz on Deepfakes | The system will display 5 images, and the user needs to classify them as real or fake | Literature Review | User | Must |
| F05 | Information about Deepfakes | The system will display an easy-to-check, manual deepfake detection | Literature Review | User | Must |
| F06 | Feedback | The system shall allow users to flag a misclassified result and submit feedback | User Testing | User | Won’t |
| | | | | | |
| N01 | Performance | The system shall display the result within 2 seconds | User Testing | System | Should |

| | | | | | |
|---|---|---|---|---|---|
| N02 | Security | The system shall not save any user-uploaded images | University Policy and GDPR | System | Must |
| N03 | Consistency | The system shall always display the same output for the same images | Best Practice | System | Should |
| N04 | Accuracy | The system shall achieve > 95% accuracy on the in-the-wild dataset | Literature Review | System | Must |
| N05 | Availability | The system shall be available online to avoid problems with installing the code and running it | Research Objective | System | Should |
| N06 | Maintainability | The system shall be updated every 6 months to include more datasets to improve the model. | User Testing | Developer | Must |
| N07 | Scalability | The system shall support 100 concurrent classifications without reducing response time. | Testing | System | Could |
| N08 | Usability | The system shall be able to help users manually identify 60 % of deepfakes after information about deepfakes. | Further Research | User | Should |

Each '**must'** requirement is derived from research questions and literature reviews. While '**should/could'** requirements are derived from following best practices and user testing. The '**won't'** requirements came from user testing. The 'won't' requirements cannot be satisfied right now due to the scope of the project and limited time restraints.

## 3.1.2 Tools and Methodologies Used -

Multiple tools and libraries were used throughout the project to preprocess images, for model training, and deployment. Each tool was selected based on previous research, its suitability for the task, and its ease of integration with **Python**, the primary programming language used. It was chosen as the primary programming language because of its extensive collection of libraries, which consist of everything from image processing to machine learning. It also works well for web deployment. Specific libraries such as NumPy, OpenCV, Matplotlib, OS and TensorFlow were used to program the application.

*3.1.2.1 Image Pre-processing -*

- **GIMP (GNU Image Manipulation Program)** – This open-source platform was used during the initial phase of the project. It helped understand the different features and find differences through smooth textures, lighting and

edge differences. It differentiated between Laplacian, Sobel and emboss and helped highlight features of fake images.

- **OpenCV and NumPy – OpenCV** was used mainly for image processing, the industry standard Python library for image processing**,** as I had mainly used this library for my placement year. They were also helpful in the batch processing of images. Image processing was done through this library to resize, filter and improve inconsistencies within images.

**Requirements Fulfilled** – F01, F05, N03, N08

*3.1.2.2 Development of Model and Testing –*

- **Kaggle** – This was used mainly to find all the datasets and understand how others have used them in their studies. Due to the ease of the application and community support, it was chosen.
- **TensorFlow and Keras** – According to Novac et al. (2022), TensorFlow performed better than PyTorch when the main priority was accuracy, and users had more control over the training flow. They also include extensive documentation and guidance to improve the networks. TensorFlow also has better compatibility with Python for implementing in a web application.
- **Grad-CAM** has been planned to be included in future work to provide explainability about the model results.

**Requirements Fulfilled** – F02, F03, N01, N04.

*3.1.2.3 Web Application –*

- **Flask** – This library was used to do web deployment.
- **Figma** – It was used to create a prototype of the web application to design it easily.
- **GitHub** – This was used for further research and collaboration opportunities.

**Requirements Fulfilled –** F04, N06.

Tools like **PyTorch** were not used because the main goal was to create an application within a short span of time due to familiarity with TensorFlow. It works better and has larger documentation and pre-trained models. GIMP was used instead of **Adobe Photoshop** because of it being open-source and works better with the research part. The option for using cloud computing has been saved for further research so that, as the dataset increased cloud computing for stronger computational skills can be used. **MATLAB** was not used because of its lower familiarity and due to complicated web deployment options.

# 3.2 Investigation

## 3.2.1 Dataset Used for Training and Evaluation

Deepfake detection models are essentially dependent on large-scale datasets for robust training and evaluation. It is extremely important for the dataset to be diverse, which helps distinguish between all the artefacts of a manipulated image.

The dataset chosen for this project consisted of and real fake images which were combined from different datasets –

1. **StyleGAN-StyleGAN2** – 2000 fake and 2000 real
2. **PDID** – 711 fake images
3. **Politics 101** – 100 real images
4. **Spitting images: tracking deepfakes and generative ai in elections** – **SITD** – 144 images

The datasets in total consisted of thousands of hundreds of images but only 5000 each for real and fake were chosen. The dataset needed to be pre-processed before use. The dataset was then divided into three parts for training, testing and validation. A split of 60-20-20 was used as suggested by Kroiß and Reschke (2025). So, 6000 images were used for training, 2000 for testing and 2000 for validation divided into equal parts of 2 for real and fake images. Th full documentation for the dataset is available in Appendix D. Decrease of demographic bias was given a priority because of which multiple datasets were used, and extra detailed analysis was carried out. The incidents dataset have been used to help the model easily identify in-the-wild images. This ensures that the model can generalise and differentiate between the images and techniques it has never seen before.

### 3.2.2 Image Processing –

The first approach identified was to take the image, then turn it into grayscale to make it easier for the machine to learn and understand the features without relying only on colours, which can be perfect in fake images. Fake images often had one thing in common: the lack of edges, inconsistent lighting and an incredibly smooth texture. Then the initial processing was using GIMP, where multiple layers of processing was experimented to identify differences between real and fake images.

It was identified that the approach of using Laplacian filtering with a kernel size of 3 demonstrated improvement for feature extraction because it was able to identify differences like lighting and depth. After this, a layer of embossing was applied, which helped enhance more facial features. This method made the image into a 3d image to help understand features better and find discrepancies quickly. Then a program was created using OpenCV and PIL to automate this process for the entire dataset by resizing images to 224x224 px, seen in many deep learning architectures, and storing the images in folders.

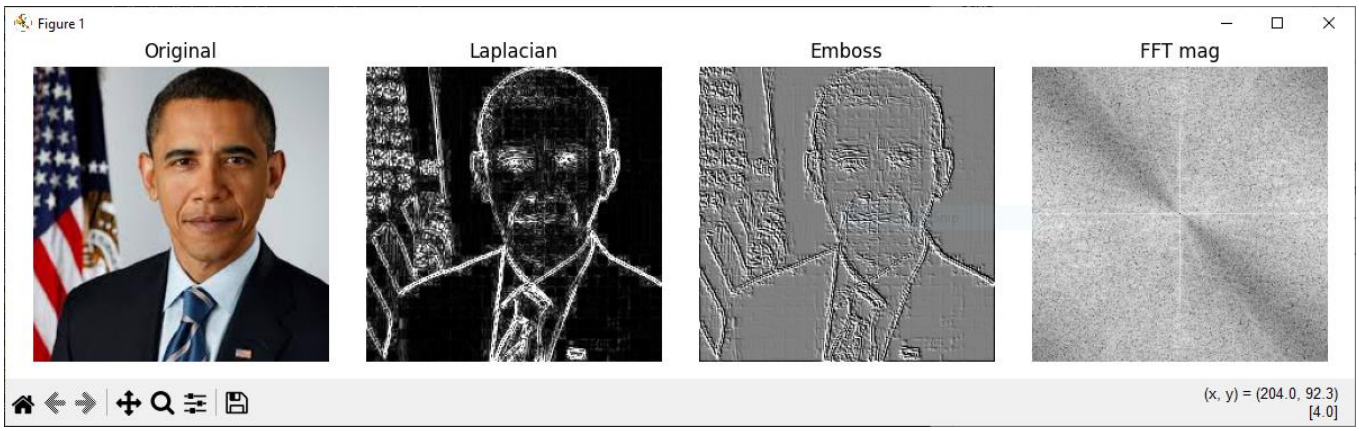

*Figure 2: Image Processing of a real image curated from Politics 101.*

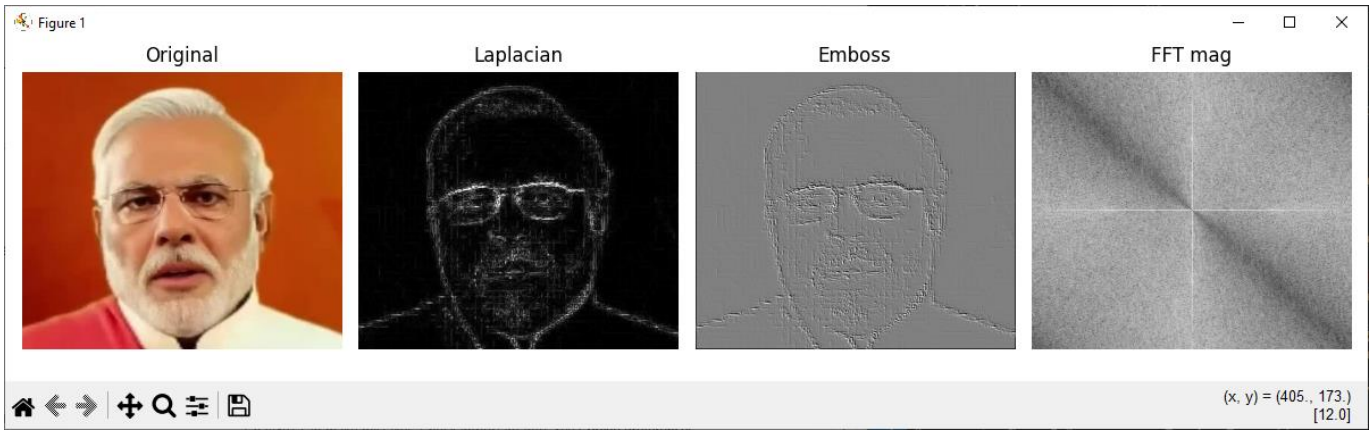


*Figure 3: Image Processing of a deepfake curated from SITD ()*

It can be easily understood from the last step that real images contain a lot of noise while the same is not available for a deepfake. Initially, histogram equalisation was used to enhance contrast in images, but it was later removed in model enhancement because it removed many features required to detect images. Then, the Fourier transform filtering was added to the process to help highlight patterns and noise that are not easily visible by simple image processing.

## 3.2.2 Machine Learning –

*3.2.2.1 Custom-built CNN* -

Multiple approaches were taken to prevent overfitting, such as random rotations, flips, and dataset compression to fit the project's context. Also, a dropout layer was added to the network. The images from the training data were augmented, such as the addition of a rotation range, which randomly rotated images up to 15 degrees, a width and height shift range, which shifted the images by up to 10%, a zoom range, which zoomed in and out of the images by up to 20%, and some were randomly flipped. This helped the model generalise better so that it prevents overfitting. Also, the brightness was changed randomly to give a compression effect, as none of the online datasets had compressed images. After preprocessing of images, it was decided through deep and thorough research to compare two architectures. Three 2D Convolution blocks were used, with 32, 64, and 128 filters respectively. Batch Normalisation was also added to stabilise training and keep activations well-scaled. The filters were doubled in every layer so the model could understand more complex features of images. When a fourth layer was added, the model started to overfit. A dropout layer was added based on good practice to drop 0.3 neurons during training, to help the model with generalisation and prevent overfitting. This custom CNN had only 100,000 parameters while ResNet50 had 24,637,313, which made it faster to train. Custom CNN also had an early stopping callback if the model did not improve for 5 epochs in a row, and a checkpoint was added, which helped save the model. All this combined helped with an analysis of the detection performance of the model.

*3.2.2.2 Fine-tuned ResNet50*

Then, through transfer learning and using Kroiß and Reschke's (2025) approach, ResNet50 was used by fine-tuning pre-trained layers to improve the model's accuracy. To prevent overfitting, multiple approaches were taken. The first was to apply data augmentation to the dataset to help train the model. Pre-trained weights of ImageNet were used, and the top layer was tuned to add a dropout layer, and GlobalAveragePooling layer. The initial layers were frozen to help the model adapt to this project's dataset. The same approach of train-test-val split was used to help train the model and understand its performance. ResNet is already trained to capture multiple features of images which our model yet cannot identify. The complete code has been backed by TensorFlow documentation (TensorFlow, n.d.).

After training both these models, the best was saved, which was later integrated into the Flask web application. Then, a Grad-CAM layer was added to help understand which features and artefacts of the images the model focused on, through heatmaps, to improve the parameters. After applying Grad-CAM, it was easy to focus on what the model was analysing and how to improve the training.

## 3.2.3 Educational Component

Initially, the approach was to create a detection model focused on compressed social media images. However, a short quiz on the web application was included to help increase awareness amongst people and help identify some easily identifiable mistakes that image generators often make. To analyse the website's workings, qualitative analysis was done by recruiting 10 participants, as suggested by Subedi (2021) (Nielsen and Levy, 1994), to recruit around 1-20 participants for qualitative research. The think-aloud protocol was used as per Nielsen (Nielsen, 2012) to help gather as much information as possible from the participants and help understand if this web application fulfilled the main requirements mentioned in the functional and non-functional requirements. Their knowledge about deepfake, their detection methods, and applications was asked both before and after the quiz. Their quiz responses were recorded and then they were shown the cues to identify deepfakes to see if their scores improved. They were also asked to use the web application via a short questionnaire, and their feedback was then used to improve the application and find out more details which would be fulfilled in future development series.

***3.2.3.1 Qualitative Analysis***

For qualitative analysis, 10 participants were recruited using a medium of Whatsapp status for user testing and analysis. They were recruited based on their rating of familiarity with social media. They were asked to complete 14 tasks which included initial checks of receipt of participant information sheet and confirmation of signed consent forms, demographic data including their age group and gender, then their knowledge of social media, deepfakes and deepfake detector was also inquired and then the quiz was conducted where five images selected randomly out of 100 total images of real and fake were shown to the participants and asked to classify them as real or fake. The images shown were random because it helped gather honest feedback from them. After the quiz, they were told hints and explained about methods various deepfake detectors used to classify images. After the hints were given, they

were then asked to take the quiz again to see if the scores improved. They were then asked to rate the layout of the web application and the ease of using the buttons and understanding the features of the application. The entire test lasted about 25 minutes in average.

## 3.3 Results –

The performance of the project is divided into two parts – quantitative evaluation through analysing the outputs of the models and qualitative evaluation through user study to analyse the usability of the education aspect of the project.

### 3.3.1 Quantitative Results –

Both models were evaluated based on which technique used for image processing was better. Accuracy, Precision, Recall and F-1 Score were compared for Laplacian+Embossing (L+E) processing and Laplacian + Embossing + FFT (L+E+F). This will allow to understand the impact of the individual image filtering techniques.

*Table 4 Comparison between image processing techniques using Custom built CNN*

| Model | Pre-processing technique used | True Positive | False Positive | True Negative | False Negative | Precision | Recall | F-1 Score | Accuracy |
|---|---|---|---|---|---|---|---|---|---|
| **Custom-CNN** | **None** | 712 | 289 | 610 | 391 | 71% | 65% | 68% | 66% |
| **Custom-CNN** | **Laplacian** | 486 | 514 | 513 | 488 | 49% | 50% | 49% | 96% |
| **Custom-CNN** | **Embossing** | 554 | 456 | 437 | 564 | 55% | 50% | 52% | 97% |
| **Custom-CNN** | **FFT** | 417 | 565 | 995 | 55 | 42% | 88% | 57% | 51% |
| **Custom CNN** | **L+E** | 315 | 62 | 439 | 186 | 84% | 63% | 72% | 72% |
| **Custom CNN** | **L+E+F** | 850 | 150 | 700 | 359 | 85% | 70% | 77% | 98% |

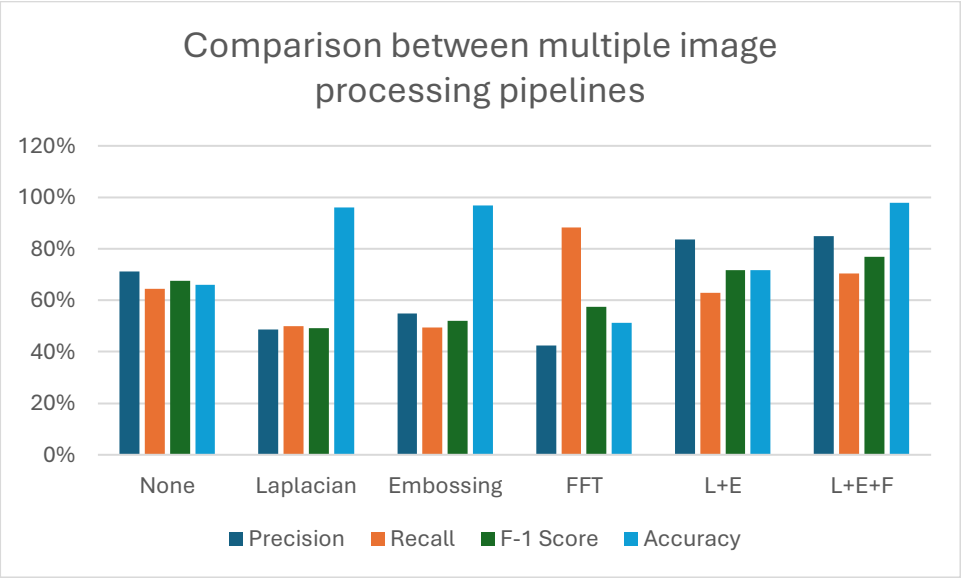


*Figure 4 Comparison between multiple image processing pipelines using Custom Built CNN*

From Table 3.1, when Custom-CNN combined with L+E+F pre-processing techniques had the highest accuracy of 97.8%, this proved this image processing pipeline to be the one which enhanced the differences between deepfake and real.

The best technique was then tested with both Custom-built CNN and ResNet50 to identify which method had better results and was quicker.

| Model | Pre-processing technique used | True Positive | False Positive | True Negative | False Negative | Precision | Recall | F-1 Score | Accuracy |
|---|---|---|---|---|---|---|---|---|---|
| **Custom CNN** | **L+E+F** | 850 | 150 | 700 | 359 | 85% | 70% | 77% | 98% |
| **Res-Net50** | **L+E+F** | 660 | 437 | 564 | 340 | 60% | 66% | 63% | 61% |

*Table 5 Comparison between CNN and ResNet50*

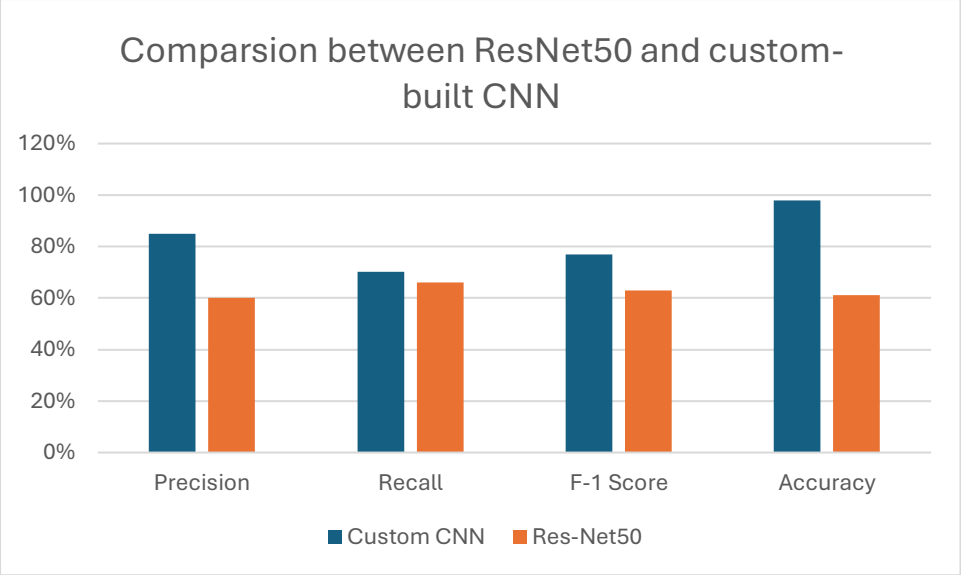


*Figure 5 Visual comparison between custom built CNN and ResNet50*

The custom CNN had 72.3% accuracy with a combination of Laplacian embossing and FFT. After FFT filtering was included, the accuracy increased to 97.8% for the custom-built CNN and 61% for ResNet50.

**Precision** was calculated using the following method –

$$\frac{True\ Positives}{True\ Positives + False\ Positives} \quad \text{----(1)}$$

**Recall** was calculated using -

$$\frac{True\ Positives}{True\ Positives + False\ Negatives} \quad \text{----(2)}$$

**F1-score** was calculated using -

$$2 * \frac{\text{Pr}ecision * Recall}{\text{Pr}ecision + Recall} \quad \text{----(3)}$$

**Confusion Matrix** – [[True Positives, False Negatives],
[False Positives, True Negatives]]

### 3.3.2 Qualitative Testing –

10 participants were recruited to check the usability of the web application, and they were sent a Participant Information Sheet and a consent form. The consent forms and participant information sheet are available in Appendix B. The participants were between 18 and 45

years old to check how well the public of all demographics understood deepfakes and could detect differences. There were 5 male and 5 female participants. All the participants had knowledge about social media and used it daily. 9 out of 10 people claimed that they had seen a deepfake of a politician recently. But only 4 said that they tried to see if it was real through google or comments of other individuals. They indicated that there was a lack of knowledge of deepfakes and deepfake detectors amongst the public and there was a need for a robust application which could identify deepfakes quickly.

**Quiz Results –**

The average score before hints were shown was 1.9 which drastically increased to 4.4. Average rating of knowledge about deepfakes was also 1.9 which increased to 8.75, similarly, knowledge of how deepfake detectors work was 1.2 which increased to 7.7 and their confidence drastically increased from 2.7 to 8.7.

**Usability –**

9 out of 10 people voted that the buttons on the website were easily visible and easy to use. 7 out of 10 people also voted that the website was also easy to use and the average score for the layout was 7.3, the most common concern was to increase the size of the font and make it more mobile friendly to help people use it on the go.

# Chapter 4

## Discussion and Evaluation of Findings –

The evaluation of the application was divided into two parts. The main part was the evaluation of the model using metrics like accuracy, F-1 Score, Precision and Grad-CAM. The second part was getting a user evaluation, which would help with the interface of the application and see how much of an educational impact it would have.

### *4.1 Quantitative Analysis –*

After training the models on two processing techniques, one included Laplacian Filtering and embossing and the other included both the techniques with Fast Fourier Transform filtering as well. The metrics calculated were Accuracy, Precision, Recall and F-1 score. The main aim of the project was to develop a model with high accuracy, which improved user awareness and used a combination of image processing with deep learning.

The model's accuracy with just images was 70% which increased to 97% with the full pipeline. This validated that the approach his project used was highly effective, especially in the context of images of politicians. Diverse datasets were used, which used multiple different techniques to produce deepfakes to help with overfitting. The most common dataset was StyleGAN-StyleGAN2 because of it being the most used deepfake generator, which resulted in a lower performance when images from different generative method was checked. However, multiple recent deepfakes were uploaded as well to help with the model's training to understand the different artefacts a deepfake generator produced and identify cues easily.

Custom CNN performed better and trained quicker than ResNet50 due to 200 times more parameters in ResNet50. Higher performance of the model when FFT was added to the image processing pipeline suggested that the inclusion of FFT helped build a robust model and method for the detection of deepfakes. ResNet50 was not chosen for the final application because it took longer to produce results, whereas a custom-built CNN took less than 2 seconds to classify an image. Also, it was kept in mind that users would be running the tool offline for the first part of the development, therefore, the custom-built CNN was a more applicable option. Using the approach of Fast Fourier Transform helped improve the accuracy scores to 97.8% after merging all three techniques together.

### *4.2 Qualitative Analysis -*

10 people were recruited to check the usability of the web application. All participants completed the quiz and the tutorial of the application. The participants average score increased by 131.58%. This suggested that if the public is shown the features which most deepfakes contain, they will be able to detect deepfake much easily and will help in increase of public trust. Participants also focused on images in more depth after they were shown the hints which helped them increase their scores. They also indicated that their knowledge of

deepfakes and deepfake detectors increased on an average of 360.53% and 541.67% respectively. They were 222.22% more confident in detecting deepfakes than they were in the beginning.  This analysis helped evaluate the need and use of public awareness of deepfake detectors and techniques.

# Chapter 5

## Conclusions and Recommendations –

### 5.1 Summary

This project aimed to develop a deepfake detection model that was tailored specifically to the current political content available on social media. A flask-based web application was created by combining multi-layered image processing through Laplacian Filtering, embossing and FFT, which was then trained by comparing a custom-made CNN and fine-tuned ResNet50, and included a quiz which helped users use this model to identify deepfakes manually and help them pinpoint easy to spot differences in real and deepfake images.

The project fulfilled its main objective of detecting images with 96% accuracy under 2 seconds per image without retaining any user data when they uploaded their images. Only Custom CNN model exceeded 96% accuracy on the in-the-wild test, which demonstrated that the multi-layered processing with neural network helped create a robust model. Through chapters 1 and 2, threats of deepfakes towards politicians, helped build a foundation and need for this project, through identifying that the existing models often failed when given compressed images. This project filled that gap, and detector was able to identify in-the-wild images found on social media of politicians and currently detect if they are deepfakes or not.

### 5.2 Key Findings

The research questions which needed to be evaluated were -

- How effective are image filtering and machine learning in detecting deepfake images in political social media posts?
- Which image filtering techniques contribute most significantly in enhancing deep fake detection accuracy of politician images?
- Which techniques are the best for enhancing detection using image filtering?

All three research questions discussed in Chapter 1 were researched, and answers were found. The best technique was a combination of all including Fast Fourier Transform, Laplacian and embossing because it had an accuracy of 97.8% when used alone, while Laplacian had 70%, and Embossing alone had 77% accuracy. Laplacian filtering and embossing, when combined with Fast Fourier Transform filtering, performed 20% better than without FFT, which suggested that all three, when combined, contributed most significantly to enhancing the accuracy of this deepfake detector. Multi-layered image processing techniques combined with deep learning enhanced details in images and the main difference which GANs were not able to replicate which was noise, was identified and tested through FFT. This project got 97.87% accuracy in detecting deepfakes of politicians from social media which had compression. CNN achieved over 95% accuracy for in-the-wild testing dataset, which suggested that it

would work well for the real-world content. ResNet50 showed better accuracy when displayed with in-the-wild images, while a custom-made CNN was quicker to train on the same datasets.

A unique perspective of this project was the inclusion of the quiz platform and increase awareness of deepfakes and helping people understand basic cues to identify deepfakes found on social media easily to prevent misinformation. The model was then uploaded to GitHub for easy access and the web application for the quiz was uploaded to NuWebSpace. The detector was able to classify an image within two seconds and did not store user images when they uploaded it. 9 out of 10 users successfully identified >60% deepfakes in the quiz after understanding the hints about the differences between deepfakes and real images. 10 out of 10 said their knowledge about deepfakes and detection techniques increased significantly.

## 5.3 Limitations

There were several limitations identified which could be addressed in future works. Even though, the CNN model had high accuracy, a more on-the-go mobile application would benefit more to the public which was beyond the context of this project but will be integrated in future works. This project only focused on images and not videos which will pose a bigger challenge in the future because deepfake videos created up until now are still easy to identify than images, but the context of videos will be integrated in future work. Even one false positive could have very serious repercussions and have a negative impact on the public trust and could potentially create misinformation. Future work should include a larger dataset for politicians to help focus on each aspect of an image and help classify easily.

## 5.4 Future Work

Future work would include the scope of videos through frame-by-frame analysis. To reduce demographic bias, the dataset would be increased to adding more images. Use of Cloud Computing and higher-powered computational resources would help store more images, train the model quicker, and analyse the problems early during the training. Integration of GradCAM to explain the deepfakes when users upload their images to help them identify the area of manipulation. Also, integration of a feedback button when user upload their images and ask for consent and saving the images when the model identifies images incorrectly to help increase the dataset, helping improve the accuracy of the model, and helping it understand more cues to identify deepfakes.

## 5.5 Reflections and Personal Development

This project had significantly developed my technical and research skills. Planning a complex project which included both image processing and deep learning. A detailed plan was created during week 7 (See Appendix A) and adjusted as the project progressed which improved my project and time management skills. My skills in deep learning and computer vision improved a lot by building a custom CNN from scratch, which helped me understand different approaches to fine-tune ResNet50 as well and enhance my understanding of neural networks. I further developed my skills in OpenCV for image processing, debugging, testing, and improving the speed of the application.

Skills like writing documentation for the code helped understand the results more thoroughly for further development as well. Understanding the importance of user testing will help me with my future projects as well. These combined skills have helped me improve my academic research and coding skills, which will be invaluable in my future endeavours.

# Reference list


1. Abdullah-Al-Wadud, M., Kabir, Md., Akber Dewan, M. and Chae, O. (2007). A Dynamic Histogram Equalization for Image Contrast Enhancement. *IEEE Transactions on Consumer Electronics*, 53(2), pp.593–600. doi:https://doi.org/10.1109/tce.2007.381734.
2. Agarwal, A. and Ratha, N. (2024). *Deepfake Catcher: Can a Simple Fusion be Effective and Outperform Complex DNNs?* [online] Available at: https://openaccess.thecvf.com/content/CVPR2024W/DFAD/papers/Agarwal_Deepfake_Catcher_Can_a_Simple_Fusion_be_Effective_and_Outperform_CVPRW_2024_paper.pdf [Accessed 15 Aug. 2024].
3. Ahmed, S.R., Sonuç, E., Ahmed, M.R. and Duru, A.D. (2022). *Analysis Survey on Deepfake detection and Recognition with Convolutional Neural Networks*. [online] IEEE Xplore. doi:https://doi.org/10.1109/HORA55278.2022.9799858.
4. Ajewole Olaitan (2024). *FAGE_v2 Dataset*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/ajewoleolaitan/fage-dataset [Accessed 5 May 2025].
5. Ajewole, F., Akinyemi, J.D., Tope, L.K. and Williams (2024). *Unmasking the Uniqueness: A Glimpse into Age-Invariant Face Recognition of Indigenous African Faces*. [online] arXiv.org. Available at: https://arxiv.org/abs/2408.06806 [Accessed 5 May 2025].
6. Anlen, S. and Llorente, R.V. (2024). *Spotting the deepfakes in this year of elections: how AI detection tools work and where they fail | Reuters Institute for the Study of Journalism*. [online] reutersinstitute.politics.ox.ac.uk. Available at: https://reutersinstitute.politics.ox.ac.uk/news/spotting-deepfakes-year-elections-how-ai-detection-tools-work-and-where-they-fail.
7. Antoniou, C. (2020). *Politics 101*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/cantonioupao/politics-101 [Accessed 5 May 2025].
8. Auburn, L. (2024). *Are video deepfakes powerful enough to influence political discourse?* [online] News Center. Available at: https://www.rochester.edu/newscenter/video-deepfakes-ai-meaning-definition-technology-623572/.
9. Ballew II, C.C. and Todorov, A. (2007). *Predicting political elections from rapid and unreflective face judgments*. [online] Available at: https://cpb-us-w2.wpmucdn.com/voices.uchicago.edu/dist/f/3051/files/2021/02/BallewTodorovPNAS2007.pdf [Accessed 5 May 2025].

10. Bell, B. and Cooney, C. (2024). Taylor Swift Vienna concerts cancelled after attack threat. *BBC News*. [online] 7 Aug. Available at: https://www.bbc.co.uk/news/articles/ce31zxqypxpo.
11. Bhargava, K. (2023). *StyleGan-StyleGan2 Deepfake Face Images*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/kshitizbhargava/deepfake-face-images.
12. Bitouk, D., Kumar, N., Dhillon, S., Belhumeur, P. and Nayar, S.K. (2008). Face swapping. *ACM SIGGRAPH 2008 papers on - SIGGRAPH '08*. doi:https://doi.org/10.1145/1399504.1360638.
13. Bond, S. (2024). *How AI deepfakes polluted elections in 2024*. [online] NPR. Available at: https://www.npr.org/2024/12/21/nx-s1-5220301/deepfakes-memes-artificial-intelligence-elections.
14. Burgess, S. (2022). *Ukraine war: Deepfake video of Zelenskyy telling Ukrainians to 'lay down arms' debunked*. [online] Sky News. Available at: https://news.sky.com/story/ukraine-war-deepfake-video-of-zelenskyy-telling-ukrainians-to-lay-down-arms-debunked-12567789.
15. Ciftci, U.A. and Demir, I. (2020). FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, [online] pp.1–1. doi:https://doi.org/10.1109/TPAMI.2020.3009287.
16. Clayton, J. (2023). *Intel's deepfake detector tested on real and fake videos*. *BBC*. Available at: https://www.bbc.co.uk/news/technology-66267961.
17. Dang-Nguyen, D.-T., Sjøen, V.V., Le, D.-H., Dao, T.-P., Tran, A.-D. and Tran, M.-T. (2023). *Practical Analyses of How Common Social Media Platforms and Photo Storage Services Handle Uploaded Images*. [online] arXiv.org. Available at: https://arxiv.org/abs/2302.12133 [Accessed 19 Apr. 2025].
18. Dave, S. (2025). *Grok: A dangerous precedent in AI-Driven misinformation*. [online] Organiser. Available at: https://organiser.org/2025/03/19/282891/world/grok-a-dangerous-precedent-in-ai-driven-misinformation/ [Accessed 1 Apr. 2025].
19. Dennehy, F. (2024). 9 in 10 concerned about deepfakes affecting election results. *The Alan Turing Institute*. [online] 2 Jul. Available at: https://www.turing.ac.uk/news/9-10-concerned-about-deepfakes-affecting-election-results.
20. Dhanuraj, D. and Solomon, N. (2024). *GENERATIVE AI AND ITS INFLUENCE ON INDIA'S 2024 ELECTIONS Prospects and Challenges in the Democratic Process*. [online] Available at: https://www.freiheit.org/sites/default/files/2025-01/a4_policy-paper_ai-on-indias-2024-electons_en-4.pdf [Accessed 1 Apr. 2025].

21. Ding, J. and Li, X. (2018). An Approach for Validating Quality of Datasets for Machine Learning. *2021 IEEE International Conference on Big Data (Big Data)*. [online] doi:https://doi.org/10.1109/bigdata.2018.8622640.
22. Eadicicco, L. (2019). *FaceApp privacy: What we know about the Russian company behind it*. [online] Business Insider. Available at: https://www.businessinsider.com/faceapp-privacy-concerns-russian-company-behind-it-2019-7#before-launching-faceapp-goncharov-worked-for-companies-such-as-yandex-and-microsoft-3 [Accessed 8 Apr. 2025].
23. Gamage, D., Ghasiya, P., Bonagiri, V., Whiting, M. and Sasahara, K. (2022). Are Deepfakes Concerning? Analyzing Conversations of Deepfakes on Reddit and Exploring Societal Implications. [online] doi:https://doi.org/10.1145/3491102.3517446.
24. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y. (2014). *Generative Adversarial Nets*. [online] Available at: https://proceedings.neurips.cc/paper_files/paper/2014/file/f033ed80deb0234979a61f95710dbe25-Paper.pdf.
25. Gorman , L., Tanner, B., Goldenberg, C., Sava, G. and Peres , R. (2024). *Spitting Images: Tracking Deepfakes and Generative AI in Elections*. [online] German Marshall Fund of the United States. Available at: https://www.gmfus.org/spitting-images-tracking-deepfakes-and-generative-ai-elections.
26. Guarnera, L., Giudice, O. and Battiato, S. (2020). *DeepFake Detection by Analyzing Convolutional Traces*. [online] Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. Available at: https://openaccess.thecvf.com/content_CVPRW_2020/papers/w39/Guarnera_DeepFake_Detection_by_Analyzing_Convolutional_Traces_CVPRW_2020_paper.pdf.
27. Guarnera, L., Giudice, O., Nastasi, C. and Battiato, S. (2020). *Preliminary Forensics Analysis of DeepFake Images*. 2020 AEIT international annual conference (AEIT) , pp.1–6.
28. Guera, D. and Delp, E.J. (2018). Deepfake Video Detection Using Recurrent Neural Networks. *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. doi:https://doi.org/10.1109/avss.2018.8639163.
29. Guo, H., Shu Fen Hu, Wang, X., Chang, M.-C. and Lyu, S. (2022). Eyes Tell All: Irregular Pupil Shapes Reveal GAN-Generated Faces. *ICASSP 2022 - 2022 IEEE*

*International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.2904–2908. doi:https://doi.org/10.1109/icassp43922.2022.9746597.

30. Hern, A. (2018). Reddit bans 'deepfakes' face-swap porn community. *The Guardian*. [online] 8 Feb. Available at: https://www.theguardian.com/technology/2018/feb/08/reddit-bans-deepfakes-face-swap-porn-community.
31. Hsu, J. (2024). *Deepfake politicians may have a big influence on India's elections*. [online] New Scientist. Available at: https://www.newscientist.com/article/2427842-deepfake-politicians-may-have-a-big-influence-on-indias-elections/.
32. Intel (2022). *Trusted Media: Real-time FakeCatcher for Deepfake Detection*. [online] Intel. Available at: https://www.intel.com/content/www/us/en/research/trusted-media-deepfake-detection.html.
33. Jacobson, N. (2024). *Deepfakes and Their Impact on Society*. [online] CPI OpenFox. Available at: https://www.openfox.com/deepfakes-and-their-impact-on-society/.
34. Karras, T., Laine, S. and Aila, T. (2019). A Style-Based Generator Architecture for Generative Adversarial Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp.4401–4410. doi:https://doi.org/10.1109/tpami.2020.2970919.
35. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. and Aila, T. (2020). Analyzing and Improving the Image Quality of StyleGAN. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. [online] Available at: https://arxiv.org/abs/1912.04958.
36. Kaustubh Dhote (2024). *Human Faces Dataset*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/kaustubhdhote/human-faces-dataset/data [Accessed 7 Apr. 2025].
37. Kawabe , A. 2, Okuyama , Y., Haga , R., Tomioka , Y. and Shin , J. (2023). A Dynamic Ensemble Selection of Deepfake Detectors Specialized for Individual Face Parts. *ProQuest*, [online] p.3932. doi:https://doi.org/10.3390/electronics12183932.
38. Ke, J. and Wang, L. (2023). DF-UDetector: An effective method towards robust deepfake detection via feature restoration. *Neural Networks*, 160, pp.216–226. doi:https://doi.org/10.1016/j.neunet.2023.01.001.
39. Korshunov, P. and Marcel, S. (2018). *DeepFakes: a New Threat to Face Recognition? Assessment and Detection*. [online] Available at: https://arxiv.org/pdf/1812.08685.

40. Kroiß, L. and Reschke, J. (2025). *Deepfake Detection of Face Images based on a Convolutional Neural Network*. [online] *arXiv.org*. Available at: https://arxiv.org/abs/2503.11389 [Accessed 14 Apr. 2025].
41. Lab, B. (2024). *Fake Face Detection*. [online] Huggingface.co. Available at: https://huggingface.co/spaces/tbvl/Fake_Face_Detection [Accessed 10 Apr. 2025].
42. Li, Y., Yang, X., Sun, P., QI, H. and Lyu, S. (2019). *Celeb-DF (v2): A New Dataset for DeepFake Forensics*. [online] Research Gate. Available at: https://www.researchgate.net/profile/Yuezun-Li/publication/336147158_Celeb-DF_A_New_Dataset_for_DeepFake_Forensics/links/5e1bec72a6fdcc28376e4548/Celeb-DF-A-New-Dataset-for-DeepFake-Forensics.pdf.
43. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K. and Galstyan, A. (2021). A Survey on Bias and Fairness in Machine Learning. *ACM Computing Surveys*, [online] 54(6), pp.1–35. doi:https://doi.org/10.1145/3457607.
44. Morse, J. (2017). *App creator apologizes for 'racist' filter that lightens users' skin tone*. [online] Mashable. Available at: https://mashable.com/article/faceapp-racism-selfie#zeUItoQB5iqI.
45. Mubarak, R., Tariq Alsboui, Alshaikh, O., Inuwa-Dutse, I., Khan, S. and Parkinson, S. (2023). A Survey on the Detection and Impacts of Deepfakes in Visual, Audio, and Textual Formats. *IEEE Access*, 11, pp.144497–144529. doi:https://doi.org/10.1109/access.2023.3344653.
46. Nguyen, T.T., Nguyen, C.M., Nguyen, D.T., Nguyen, D.T. and Nahavandi, S. (2019). Deep Learning for Deepfakes Creation and Detection. *arXiv:1909.11573 [cs, eess]*. [online] Available at: https://arxiv.org/abs/1909.11573.
47. Nielsen, J. (2012). *Thinking Aloud: The #1 Usability Tool*. [online] Nielsen Norman Group. Available at: https://www.nngroup.com/articles/thinking-aloud-the-1-usability-tool/.
48. Nielsen, J. and Levy, J. (1994). Measuring usability: preference vs. performance. *Communications of the ACM*, 37(4), pp.66–75. doi:https://doi.org/10.1145/175276.175282.
49. Novac, O.-C., Chirodea, M.C., Novac, C.M., Bizon, N., Oproescu, M., Stan, O.P. and Gordan, C.E. (2022). Analysis of the Application Efficiency of TensorFlow and PyTorch in Convolutional Neural Network. *Sensors*, 22(22), p.8872. doi:https://doi.org/10.3390/s22228872.
50. Olivola, C.Y., Sussman, A.B., Tsetsos, K., Kang, O.E. and Todorov, A. (2012). Republicans Prefer Republican-Looking Leaders. *Social Psychological and*

*Personality Science*, 3(5), pp.605–613. doi:https://doi.org/10.1177/1948550611432770.

51. Olivola, C.Y. and Todorov, A. (2010). Elected in 100 milliseconds: Appearance-Based Trait Inferences and Voting. *Journal of Nonverbal Behavior*, 34(2), pp.83–110. doi:https://doi.org/10.1007/s10919-009-0082-1.
52. ondyari (2022). *FaceForensics++: Learning to Detect Manipulated Facial Images*. [online] GitHub. Available at: https://github.com/ondyari/FaceForensics.
53. OpenAI (2024). *DALL·E 3*. [online] Openai.com. Available at: https://openai.com/index/dall-e-3/.
54. P, S. and Sk, S. (2021). *DeepFake Creation and Detection:A Survey*. [online] IEEE Xplore. doi:https://doi.org/10.1109/ICIRCA51532.2021.9544522.
55. Pál, M., Banjanin, B., Dedijer, S., Vladić, G. and Bošnjaković, G. (2020). Preliminary analysis of image processing-based evaluation of embossing quality. *Proceedings - The Tenth International Symposium GRID 2020*, [online] pp.269–279. doi:https://doi.org/10.24867/grid-2020-p29.
56. Paris, S., Hasinoff, S. and Kautz, J. (2015). Local Laplacian Filters: Edge-aware Image Processing with a Laplacian Pyramid. *Communications of the ACM*, [online] 58(3). Available at: https://people.csail.mit.edu/hasinoff/pubs/ParisEtAl11-lapfilters-lowres.pdf.
57. Petrov , I., Daiheng, G., Liu, K., Marangonda, S., Ume, C., Jiang, J., Rp , L., Zhang , S., Wu, P. and Zhang, W. (2021). *DeepFaceLab: Integrated, flexible and extensible face-swapping framework*. [online] Available at: https://arxiv.org/pdf/2005.05535.
58. Rehman, S.U. (2024). *Detect AI-Generated Faces: High-Quality Dataset*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/shahzaibshazoo/detect-ai-generated-faces-high-quality-dataset [Accessed 7 Apr. 2025].
59. Robins-Early, N. (2024). *How did Donald Trump end up posting Taylor Swift deepfakes?* [online] the Guardian. Available at: https://www.theguardian.com/technology/article/2024/aug/24/trump-taylor-swift-deepfakes-ai.
60. Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J. and Nießner, M. (2018). FaceForensics: A Large-scale Video Dataset for Forgery Detection in Human Faces. *arXiv:1803.09179 [cs]*. [online] Available at: https://arxiv.org/abs/1803.09179.
61. SaadAli5 (2024). *Imran Khan Images 512x512*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/saadali5/imran-khan-images-512x512/data [Accessed 5 May 2025].

62. Sahu, A.K. (2022). *Leaders Image Dataset (2022)*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/anandkumarsahu09/quad-leaders-image-dataset-2022 [Accessed 5 May 2025].
63. Sebastian, M. (2024). AI and deepfakes blur reality in India elections. *BBC News*. [online] 16 May. Available at: https://www.bbc.co.uk/news/world-asia-india-68918330.
64. singh, saksham (2021). *Indian Politicians*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/sakshamsingh1904/indian-politicians [Accessed 5 May 2025].
65. Sippy, T., Enock, F.E., Bright, J. and Margetts, H.Z. (2024). *Behind the Deepfake: 8% Create; 90% Concerned Surveying public exposure to and perceptions of deepfakes in the UK*. [online] London, UK: The Alan Turing Institute. Available at: https://www.turing.ac.uk/sites/default/files/2024-07/behind_the_deepfake_full_publication.pdf.
66. Song, H. (2023). K-Politician. [online] Kaggle. Available at: https://www.kaggle.com/dsv/6560418.
67. Subedi, K.R. (2021). Determining the sample in qualitative research. *Scholars' Journal*, 4(2645-8381), pp.1–13. doi:https://doi.org/10.3126/scholars.v4i1.42457.
68. Sudarshana, K. and Vamsidhar, Y. (2025). UAM-Net: Robust Deepfake Detection Through Hybrid Attention Into Scalable Convolutional Network. *Expert Systems*, 42(3). doi:https://doi.org/10.1111/exsy.70009.
69. Tahir, R., Batool, B., Jamshed, H., Jameel, M., Anwar, M., Ahmed, F., Zaffar, M.A. and Zaffar, M.F. (2021). Seeing is Believing: Exploring Perceptual Differences in DeepFake Videos. *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. doi:https://doi.org/10.1145/3411764.3445699.
70. Tchaptchet, E., Elie Fute Tagne, Acosta, J., Danda, R. and Kamhoua, C. (2025). Deepfakes Detection By Iris Analysis. *IEEE Access*. [online] doi:https://doi.org/10.1109/access.2025.3527868.
71. TensorFlow (n.d.). *Image classification | TensorFlow Core*. [online] TensorFlow. Available at: https://www.tensorflow.org/tutorials/images/classification.
72. Todorov, A., Mandisodza, A.N., Goren, A. and Hall, C.C. (2005). Inferences of Competence from Faces Predict Election Outcomes. *Science* , [online] 308(1623), pp.1623–1626. doi:https://doi.org/10.1126/science.1110589.

73. Varsha, P.S. (2023). How can we manage biases in artificial intelligence systems – A systematic literature review. *International Journal of Information Management Data Insights*, [online] 3(1), p.100165. doi:https://doi.org/10.1016/j.jjimei.2023.100165.
74. Venkataramakrishnan, S. (2019). *Can you believe your eyes? How deepfakes are coming for politics*. [online] www.ft.com. Available at: https://www.ft.com/content/4bf4277c-f527-11e9-a79c-bc9acae3b654.
75. Wakefield, J. (2022). Deepfake presidents used in Russia-Ukraine war. *BBC News*. [online] 18 Mar. Available at: https://www.bbc.co.uk/news/technology-60780142.
76. Walker, C.P., Schiff, D.S. and Schiff, K.J. (2024). Merging AI Incidents Research with Political Misinformation Research: Introducing the Political Deepfakes Incidents Database. In: *Arxiv.org*. [online] to policy taskforces, platform regulation, media literacy efThe Thirty-Eighth AAAI Conference on Artificial Intelligence (AAAI-24). Available at: https://arxiv.org/html/2409.15319v1 [Accessed 14 Jan. 2025].
77. Yosinski, J., Clune, J., Bengio, Y. and Lipson, H. (2014). How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, [online] 27. Available at: https://proceedings.neurips.cc/paper_files/paper/2014/hash/532a2f85b6977104bc93f8580abbb330-Abstract.html [Accessed 14 Apr. 2025].
78. Zeinabartaill (2021). *Presidents*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/zeinabartaill/presidents [Accessed 5 May 2025].

# Bibliography


1. https://learning.oreilly.com/library/view/opencv-computer-vision/9781787125490/ch06s06.html
2. https://www.tensorflow.org/tutorials/images/classification
3. https://opencv.org/
4. https://www.w3schools.com/
5. https://www.tensorflow.org/guide/

# Appendix

## Appendix A – TOR Document - KV6013 - TOR - W21037581.docx

## Appendix B – Code Snippets –

```
img = cv2.resize(img, output_size)
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

laplacian = cv2.Laplacian(gray, cv2.CV_64F)
laplacian = cv2.normalize(laplacian, None, 0, 255, cv2.NORM_MINMAX)
```

```
kernel = np.array([[-2, -1, 0],
                   [-1, 1, 1],
                   [0, 1, 2]])
embossed_image = cv2.filter2D(laplacian, -1, kernel)
```

```
f = np.fft.fft2(embossed)
fshift = np.fft.fftshift(f)
magnitude_spectrum = 20 * np.log(np.abs(fshift) + 1e-9)
magnitude_spectrum = cv2.normalize(magnitude_spectrum, None, 0, 255, cv2.NORM_MINMAX)
```

## Appendix C –

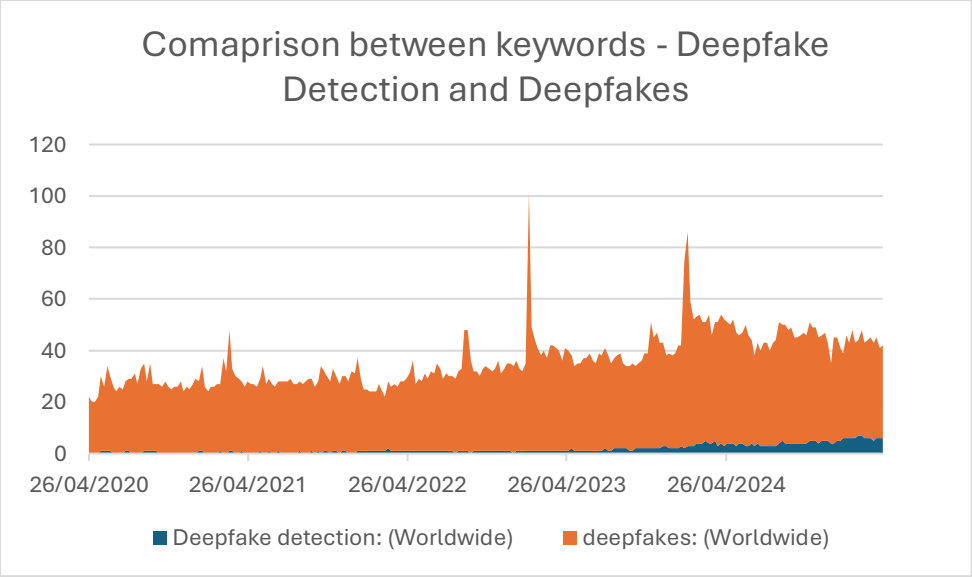


*Figure 6 Comparison between keyword trends of Deepfakes vs Deepfake Detection through Google Trends*

## Appendix D –

| **Continent** | **Country** | **Number of Real Images** |
|---|---|---|
| **Asia** | **Total** | 753 |
| | India | 300 |
| | Russia | 47 |

| | | |
|---|---|---|
| | Pakistan | 47 |
| | Korean | 300 |
| | China | 10 |
| | Japan | 49 |
| **Africa** | **Total** | **510** |
| | Algeria | |
| | Angola | |
| | Democratic Republic of the Congo | |
| | Eswatini | |
| | Ethiopia | |
| | Ghana | |
| | Kenya | |
| | Namibia | |
| | Nigeria | |
| | South Africa | |
| | Sudan | |
| | Tanzania | |
| | Uganda | |
| | Zambia | |
| | Zimbabwe | |
| **North America** | **Total** | 350 |
| | America | 350 |
| **Europe** | **Total** | 298 |
| | Italy | 73 |
| | UK | 50 |
| | Germany | 46 |
| | Spain | 42 |
| | France | 34 |
| | Greece | 53 |
| **Australia** | Australia | 80 |

More In-depth information is available on – Dataset