Universidad Politécnica
de Madrid

Escuela Técnica Superior de
Ingenieros Informáticos

Master's Programme in Digital Innovation:
Data Science (EIT Digital Master School)

Master Thesis

Deep Fake Detection: Evaluation of Several Face Forgery Detectors

Author: Víctor Morcuende Castell

Madrid, August, 2024

This Master Thesis has been deposited in ETSI Informáticos from Universidad Politécnica de Madrid.

Master Thesis

Master's Programme in Digital Innovation: Data Science (EIT Digital Master School)

Title: Deep Fake Detection: Evaluation of Several Face Forgery Detectors

August 2024

Author: Víctor Morcuende Castell

Supervisor:
Simon Malinowski

Professor and responsible for M2 Miage Data Science master

ISTIC Faculty

Institute for Research in Computer Science and Random Systems (IRISA) research center: https://www.irisa.fr/en/

Université de Rennes 1

Co-supervisor:
Marta Patiño Martínez

Professor and responsible for Master's Programme in Digital Innovation: Data Science

E.T.S DE INGENIEROS INFORMÁTICOS

Universidad Politécnica de Madrid

Abstract

This master thesis investigates the growing challenge of deepfake videos and the efficacy of various state-of-the-art detection methods in identifying such synthetic media. With the rapid advancement of artificial intelligence and machine learning techniques, deepfakes have become increasingly sophisticated, posing significant threats in areas such as misinformation, privacy, and security. The primary focus of this research is to conduct a comprehensive evaluation of four prominent deepfake detection methods—FaceForensics++, LipForensics, ID-Reveal, and POI-Forensics—by examining their performance across different datasets, video compression levels, and manipulation techniques.

Regarding FaceForensics++, this benchmark provides the foundation for this study, offering a detailed analysis of facial reenactment methods like Face2Face and NeuralTextures. While FaceForensics++ has set a standard in the evaluation of facial manipulations, it demonstrates certain vulnerabilities when exposed to compressed videos and real-world scenarios, where inconsistencies in detection accuracy are observed. These findings underscore the necessity for more robust methods that can maintain high detection rates even under challenging conditions.

Moreover, LipForensics introduces a novel approach to detecting deepfakes by focusing on lip-syncing manipulations, which are often overlooked by traditional detection methods. The evaluation conducted in this thesis highlights the effectiveness of LipForensics in scenarios where the mouth region is critical, such as in news broadcasts or interviews, where subtle discrepancies in lip movements can indicate forgery. However, the performance of LipForensics is not without limitations, as its accuracy diminishes with lower video quality, higher compression levels and specifically in cases where the mouth region is occluded, suggesting a need for further refinement in its detection algorithm.

ID-Reveal and POI-Forensics represent the next generation of deepfake detection technologies, emphasizing identity verification and multi-modal analysis, respectively. ID-Reveal leverages the unique characteristics of an individual's facial identity to detect discrepancies introduced by deepfake manipulations. This method proves particularly effective in high-resolution, controlled environments but faces challenges in more dynamic, low-quality video scenarios. POI-Forensics, on the other hand, incorporates multi-modal detection strategies, analyzing not only visual cues but also audio and behavioral patterns to detect deepfakes. The comprehensive nature of POI-Forensics enables it to outperform the other methods in detecting manipulations in compressed videos, which is increasingly relevant as deepfakes are often distributed in formats that degrade video quality.

The research presented in this thesis was conducted under certain constraints, particularly concerning the time available for evaluation and the size of the test datasets. While these limitations have restricted the scope of the analysis, the findings provide significant insights into the current state of deepfake detection technologies. The study reveals that while each method has its strengths, no single approach offers a complete solution to the deepfake detection challenge, particularly when faced with the variability of real-world conditions, such as compression artifacts and diverse manipulation techniques.

This work contributes to the field by not only assessing the effectiveness of existing detection methods, but also by identifying areas where future research is needed. It suggests that advancements in deepfake detection will likely come from a combination of techniques, potentially integrating multiple modalities and employing more sophisticated machine learning models that can adapt to the evolving nature of synthetic media. Furthermore, the study appeals to the development of more comprehensive datasets that better reflect the complexities of real-world applications, which will be essential for training and evaluating the next generation of deepfake detection systems.

In conclusion, this master thesis highlights the critical importance of developing robust, adaptable, and scalable deepfake detection methods as the threat posed by synthetic media continues to grow. The insights gained from this research provide a foundation for future work aimed at enhancing the detection of deepfakes, thereby contributing to the broader effort to safeguard the integrity of digital media in an era of technological advancement.

Table of Contents

List of Tables

List of Figures

1 Introduction

The arrival of advanced machine learning and computer vision technologies has conducted us to an era of unprecedented creativity and innovation, yet simultaneously, it has introduced challenges that demand our vigilant attention. One such challenge is the rising popularity of deepfake content: synthetic media that harnesses artificial intelligence to manipulate or fabricate visual and auditory elements, often convincingly mimicking real human expressions and actions. The implications of deepfakes extend far beyond the realm of entertainment, as they present a powerful tool for the dissemination of misinformation, identity theft, and malicious activities. As a result, the urgency to develop robust methodologies against deepfake has become a fundamental concern in the fields of data science, cybersecurity, media forensics, and beyond.

This work delves into the field of deepfake detection, describing the techniques employed to discern authentic content from its synthetic counterparts. The significance of this attempt lies in safeguarding the integrity of digital content and strengthening our defenses against deceptive practices that undermine the trustworthiness of visual and auditory information. As technology continues to evolve, so do the capabilities of those who seek to exploit it for malicious purposes. Therefore, the search for effective deepfake detection mechanisms assumes a crucial role in mitigating these potential harms.

In this work, we navigate through the complexities of face forgery detection, explaining the methodologies and strategies employed to distinguish genuine facial expressions from manipulated ones. Moreover, we explore the development and refinement of detection algorithms, which represent a proactive response to the challenges posed by deepfakes.

Facial manipulation techniques within the realm of deepfake content can be broadly categorized into two distinct types: facial identity manipulation (or face swapping) and facial expression manipulation (also known as face reenactment).

Facial identity manipulation techniques involve the seamless replacement of one person's face with another's in video or image content. The rise of face swapping applications has sparked concerns due to their potential misuse for malicious activities, such as impersonation, identity theft, or the creation of misleading content. On the other hand, facial expression manipulation techniques focus on manipulating existing facial expressions within a video by superimposing them onto a target individual in real-time. These techniques often require sophisticated algorithms to accurately capture and replicate the subtle movements and expressions of the source face onto the destination face. Unlike the first type, facial expression manipulation techniques do not alter the overall identity of the target individual, but rather modifies their expressions or gestures in a convincingly deceptive manner.

Figure 1: Facial identity manipulation example

Figure 2: Facial expression manipulation

As a result, the focus of this work is mainly on face reenactment methods, since they introduce subtler alterations that can be more challenging to detect, in comparison with the widely recognized face swapping techniques. The complexity of face reenactment techniques raises the stakes in the ongoing efforts to develop effective detection mechanisms, making it a relevant area of exploration for this work.

Consequently, this report takes a comprehensive approach to explain the main methods employed in face reenactment, providing a detailed overview of these techniques, and conducting a thorough evaluation which highlights their strengths and limitations. Additionally, the report presents an evaluation of the results obtained from each algorithm as well as a comparison between them.

2 Deep Fake Detection Approaches

2.1 FaceForensics++

2.1.1 Overview

FaceForensics++ [1] has significantly contributed to the advancement of deepfake detection through its innovative benchmark, automated evaluation protocols, and a vast dataset that exploits various facial manipulation techniques. The meticulously designed benchmark incorporates methods like *DeepFakes*, **Face2Face** (F2F), *FaceSwap*, and **NeuralTextures** (NT) to create a standardized and comprehensive evaluation framework. It is publicly accessible and encompasses over 1.8 million manipulated images, presenting a considerable scale for evaluating forgery detection algorithms.

This dataset surpasses other publicly available forgery datasets in scale and diversity, enhancing its relevance and utility for deepfake detection research. Moreover, it comprises 1,000 videos with pristine sources obtained from YouTube, which have undergone a meticulous screening process to ensure high-quality selections, including front facing target faces to prevent failures.

Furthermore, the dataset involves several face manipulation methods, however, this report focuses on the face reenactment ones: Face2Face (computer graphics-based) and NeuralTextures (learning-based).

Face2Face is a facial expression transfer system designed to maintain the identity of the target person. It relies on automatic keyframe selection in two video input streams to reconstruct the face and then re-synthesize it with different expressions and lighting conditions. This process involves a preprocessing step that uses initial frames to establish a temporary face identity (3D model) and tracks expressions throughout the video. Automatic selection of keyframes is then performed to transfer source expression parameters to the target video, generating reenactment outputs.

NeuralTextures, on the opposite, employs original video data to learn a neural texture of the target person, along with a rendering network. Training involves photometric reconstruction and adversarial losses, as well as Generative Adversarial Network (GAN). The approach focuses on modifying facial expressions in the mouth region while keeping the eye region unchanged for realism. Additionally, it includes post-processing steps to simulate video compression and generate videos with varying quality levels, resembling those on social networks.

Figure 3: Example of Face2Face and NeuralTextures face manipulation methods (Source: FaceForensics++ paper [1])

In the postprocessing phase, the manipulated videos undergo different video quality adjustments to add variety, resulting in several quality levels: first, raw videos (c0), using the H.264 codec, which is widely used by social networks or video sharing websites. Then, high-quality (HQ) videos (c23), nearly lossless with a constant rate quantization parameter of 23, and finally, low-quality (LQ) videos (c40), with a quantization of 40.

The forgery detection task is framed as a per-frame binary classification problem for manipulated videos. Also, to enhance this process, domain-specific information is incorporated, involving the use of face tracking methods to extract the face region of the image (instead of utilizing the entire image), and hence improve the forgery detection performance.

Another key point is the network architecture *XceptionNet*, a traditional CNN trained on ImageNet with separable convolutions and residual connections that stands out by achieving superior results in detecting manipulated videos. The network undergoes pre-training and subsequent fine-tuning, demonstrating robust performance across varying compression levels and video quality. Moreover, the results indicate that automated detectors not only outperform human observers, but also other methods, especially when combining XceptionNet with the domain-specific information.

2.1.2 Performed Work

After describing in detail the mechanisms and fundamental principles of the FaceForensics++ method, this section explains the main activities performed on this dataset, available on GitHub [2]. It captures the activities undertaken, the achieved results, the decisions taken, and any encountered challenges.

The primary aim was to replicate a scaled-down version of the FaceForensics++ dataset, considering our storage limitations, computational and time constraints. The focus was on assessing its efficacy in detecting deepfake content, with a subsequent comparison of results against the paper's findings.

Therefore, the initial step to detect deepfake content involved acquiring the required videos (original and manipulated) for testing the face reenactment methods (Face2Face and NeuralTextures). To streamline the process, it was decided to extract only 2 out of the 3 video quality levels: the raw videos (c0) and the low-quality ones (c40). This decision was based on the reasoning that covering both, the high and the low ends of video quality levels with the chosen pairs would avoid the need for the high-quality (c23) videos.

Following this, the accuracy of their best method, *XceptionNet*, was tested using a face detection binary classification (either "real" or "fake") network. Moreover, the corresponding pretrained models and weights, already available for each quality-level video (that is, for the c0 and c40 compression videos), were utilized.

To begin with the evaluation, the test set from the dataset was downloaded, which was made up of a fixed training, validation, and test sets split (720, 140, and 140 videos, respectively). After that, a Python script ("extract_datasets_from_videos") was created to automatically extract the desired test set videos from the split mentioned in the paper.

Figure 4: First frame of the first video from the original videos test set. Raw quality (c0) on the left and low quality (c40) on the right

Then, to properly evaluate the test set videos, another Python script ("detect_from_video") was employed to analyze videos and determine whether the faces within them were classified as "real" or "fake" using the *XceptionNet* architecture. The mentioned script processed each frame of the input video by applying a face detection mechanism to identify faces within the frames. For each detected face, the script extracted a specific region of the frame and passed it through the *XceptionNet* model for classification. The result or prediction (whether the face was considered "real" or "fake"), was then superimposed onto the original video frame along with a bounding box around the detected face, offering an intuitive way to observe the model's classification outcomes.

Figure 5: First frame of the first video from the original (upper), Face2Face (left) and NeuralTextures (right) videos, all with raw quality. The bounding boxes and predictions can be appreciated

However, to enhance the interpretability of the results, the code underwent modifications to provide a more detailed and informative analysis of the videos as well as understanding of the model's performance. As a result, a clip-level prediction and accuracy calculation were introduced, in which, for each video clip, the average probability for both the "real" and "fake" classes was computed. This was achieved by aggregating predictions from individual frames (of the same video), offering insights into the overall characteristics of the video content.

Additionally, a new accuracy measure was introduced, comparing the average probabilities with the inferred ground truth label, given by nature of the videos (either original/real or manipulated/fake). Furthermore, the script was able to calculate the total accuracy across all videos in the specified folder (Face2Face folder, for example), providing a comprehensive evaluation of the model's performance on a broader scale. In this way, these novel modifications enable a detailed assessment of each face reenactment method, including pristine videos.

Nevertheless, during the evaluation of the test set videos, we encountered an issue with the processing time, as it was excessively long for each video (below it can be appreciated a chart proving this). Given that the evaluation script had to analyze every frame to detect and classify faces, this extended processing time became impractical. Subsequently, a strategic decision was made to truncate the test set and assess only the first 20 videos. This adjustment was uniformly applied across all methods, including Face2Face, NeuralTextures, and the original sequences, ensuring a fair and consistent comparison among them.

Regarding our dataset, we chose to expand it with additional "in the wild" videos, both from pristine and manipulated sequences. This decision was motivated by the paper's acknowledgment that their initial attempt to include such videos faced challenges due to the requirement for the target face to be front facing. Our aim was to assess the performance of the face reenactment methods and the *XceptionNet* model in scenarios beyond the FaceForensics++ dataset. To this end, we collected a set of 5 "in the wild" original videos from YouTube ([3]-[7]) and 6 manipulated counterparts from the same source ([8]-[13]). Like with the previous dataset, we applied the evaluation script to assess these videos, facilitating a comparative analysis of the outcomes with those obtained from FaceForensics++.

2.1.3 Results

This section offers a comprehensive exploration into the performance of face forgery detection methods across various compressions and models. Through meticulous evaluation, we explore the details of the distinct methods seen and the analysis of manipulated, original and "in the wild" videos. By examining the outcomes under different conditions, we aim to discern patterns and reveal key insights about these algorithms.

Face2Face method – Raw (c0) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_003.mp4	1.139e-22	1.0	Fake	100%
003_000.mp4	2.086e-23	1.0	Fake	100%
012_026.mp4	2.843e-22	1.0	Fake	100%
015_919.mp4	9.577e-25	1.0	Fake	100%
024_073.mp4	2.546e-07	0.999	Fake	100%
026_012.mp4	1.527e-18	1.0	Fake	100%
029_048.mp4	1.125e-22	1.0	Fake	100%
035_036.mp4	1.705e-22	1.0	Fake	100%
036_035.mp4	1.137e-29	1.0	Fake	100%
044_945.mp4	2.564e-09	0.999	Fake	100%
047_862.mp4	1.479e-26	1.0	Fake	100%
048_029.mp4	0.007924	0.992076	Fake	100%
073_024.mp4	2.404e-08	0.999	Fake	100%
078_955.mp4	2.370e-19	1.0	Fake	100%
102_114.mp4	3.988e-12	1.0	Fake	100%
114_102.mp4	1.352e-30	1.0	Fake	100%
128_896.mp4	5.849e-15	1.0	Fake	100%
135_880.mp4	1.649e-24	1.0	Fake	100%
138_142.mp4	1.628e-26	1.0	Fake	100%
141_161.mp4	7.731e-24	1.0	Fake	100%
Total Accuracy				**100.0%**

Table 1: Face2Face method – Raw (c0) compression – Raw (c0) model

Face2Face method – LQ (c40) compression – LQ (c40) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_003.mp4	0.273	0.727	Fake	100%
003_000.mp4	0.018	0.982	Fake	100%
012_026.mp4	0.573	0.427	Fake	**0%**
015_919.mp4	0.285	0.715	Fake	100%
024_073.mp4	0.04008	0.95992	Fake	100%
026_012.mp4	0.409	0.591	Fake	100%
029_048.mp4	0.076	0.924	Fake	100%
035_036.mp4	0.435	0.565	Fake	100%
036_035.mp4	0.207	0.793	Fake	100%
044_945.mp4	0.6698	0.3302	Fake	**0%**
047_862.mp4	0.032	0.968	Fake	100%
048_029.mp4	0.453	0.547	Fake	100%
073_024.mp4	0.412	0.588	Fake	100%
078_955.mp4	0.609	0.390	Fake	**0%**
102_114.mp4	0.376	0.624	Fake	100%
114_102.mp4	0.279	0.721	Fake	100%
128_896.mp4	0.352	0.648	Fake	100%
135_880.mp4	0.133	0.867	Fake	100%
138_142.mp4	0.004	0.996	Fake	100%
141_161.mp4	0.615	0.385	Fake	**0%**
Total Accuracy				**80.0%**

Table 2: Face2Face method – LQ (c40) compression – LQ (c40) model

Face2Face method – LQ (c40) compression – HQ (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_003.mp4	0.987	0.013	Fake	0%
003_000.mp4	0.981	0.019	Fake	0%
012_026.mp4	0.844	0.156	Fake	0%
015_919.mp4	0.937	0.063	Fake	0%
024_073.mp4	0.99	0.01	Fake	0%
026_012.mp4	0.928	0.072	Fake	0%
029_048.mp4	0.971	0.029	Fake	0%
035_036.mp4	0.994	0.006	Fake	0%
036_035.mp4	0.972	0.028	Fake	0%
044_945.mp4	0.991	0.009	Fake	0%
047_862.mp4	0.554	0.446	Fake	0%
048_029.mp4	0.987	0.013	Fake	0%
073_024.mp4	0.977	0.023	Fake	0%
078_955.mp4	0.9995	0.0005	Fake	0%
102_114.mp4	0.995	0.005	Fake	0%
114_102.mp4	0.976	0.024	Fake	0%
128_896.mp4	0.622	0.378	Fake	0%
135_880.mp4	0.738	0.262	Fake	0%
138_142.mp4	0.992	0.008	Fake	0%
141_161.mp4	0.914	0.086	Fake	0%
Total Accuracy				**0.0%**

Table 3: Face2Face method – LQ (c40) compression – HQ (c0) model

In evaluating the Face2Face method, it becomes evident that the raw (c0) compression paired with the raw (c0) model yields faultless results, correctly classifying all videos as fake with high confidence. However, when tested on lower-quality compressions, specifically LQ (c40) compression with HQ (c0) models, the accuracy takes a significant hit. This decline suggests that the Face2Face method might struggle when confronted with compressed and lower-quality input, emphasizing the importance of raw quality data for optimal performance.

NeuralTextures method – Raw (c0) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_003.mp4	2.884e-10	0.999	Fake	100%
003_000.mp4	0.855	0.145	Fake	**0%**
012_026.mp4	0.00024	0.99976	Fake	100%
015_919.mp4	4.129e-06	0.99999587	Fake	100%
024_073.mp4	0.00651	0.993489	Fake	100%
026_012.mp4	0.01058	0.98942	Fake	100%
029_048.mp4	3.244e-09	0.9999999974	Fake	100%
035_036.mp4	0.007	0.993	Fake	100%
036_035.mp4	0.00032	0.99968	Fake	100%
044_945.mp4	0.15	0.85	Fake	100%
047_862.mp4	0.4461	0.5539	Fake	100%
048_029.mp4	0.9967	0.0033	Fake	**0%**
073_024.mp4	0.992	0.00783	Fake	**0%**
078_955.mp4	2.827e-07	0.9999997	Fake	100%
102_114.mp4	0.445	0.555	Fake	100%
114_102.mp4	0.0348	0.9652	Fake	100%
128_896.mp4	0.002	0.998	Fake	100%
135_880.mp4	0.185	0.815	Fake	100%
138_142.mp4	3.293e-13	1.0	Fake	100%
141_161.mp4	0.0027	0.9973	Fake	100%
Total Accuracy				**85.0%**

Table 4: NeuralTextures method – Raw (c0) compression – Raw (c0) model

NeuralTextures method – LQ (c40) compression – LQ (c40) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_003.mp4	0.5008	0.4992	Fake	0%
003_000.mp4	0.4804	0.5196	Fake	**100%**
012_026.mp4	0.692	0.308	Fake	0%
015_919.mp4	0.422	0.578	Fake	**100%**
024_073.mp4	0.37	0.63	Fake	**100%**
026_012.mp4	0.883	0.117	Fake	0%
029_048.mp4	0.152	0.848	Fake	**100%**
035_036.mp4	0.7006	0.2994	Fake	0%
036_035.mp4	0.777	0.223	Fake	0%
044_945.mp4	0.966	0.034	Fake	0%
047_862.mp4	0.507	0.493	Fake	0%
048_029.mp4	0.968	0.032	Fake	0%
073_024.mp4	0.821	0.179	Fake	0%
078_955.mp4	0.9977	0.0023	Fake	0%
102_114.mp4	0.715	0.285	Fake	0%
114_102.mp4	0.597	0.403	Fake	0%
128_896.mp4	0.572	0.428	Fake	0%
135_880.mp4	0.9999997	2.939e-07	Fake	0%
138_142.mp4	0.816	0.184	Fake	0%
141_161.mp4	0.777	0.223	Fake	0%
Total Accuracy				**20.0%**

Table 5: NeuralTextures method – LQ (c40) compression – LQ (c40) model

NeuralTextures method – LQ (c40) compression – HQ (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_003.mp4	0.9992	0.0008	Fake	0%
003_000.mp4	0.992	0.008	Fake	0%
012_026.mp4	0.85	0.15	Fake	0%
015_919.mp4	0.823	0.177	Fake	0%
024_073.mp4	0.994	0.006	Fake	0%
026_012.mp4	0.922	0.078	Fake	0%
029_048.mp4	0.979	0.021	Fake	0%
035_036.mp4	0.984	0.016	Fake	0%
036_035.mp4	0.978	0.022	Fake	0%
044_945.mp4	0.998	0.002	Fake	0%
047_862.mp4	0.851	0.149	Fake	0%
048_029.mp4	0.97	0.03	Fake	0%
073_024.mp4	0.994	0.006	Fake	0%
078_955.mp4	0.9998	0.0002	Fake	0%
102_114.mp4	0.986	0.014	Fake	0%
114_102.mp4	0.98	0.02	Fake	0%
128_896.mp4	0.709	0.291	Fake	0%
135_880.mp4	0.991	0.009	Fake	0%
138_142.mp4	0.991	0.009	Fake	0%
141_161.mp4	0.87	0.13	Fake	0%
Total Accuracy				**0.0%**

Table 6: NeuralTextures method – LQ (c40) compression – HQ (c0) model

The NeuralTextures method exhibits a distinctive behavior. When utilizing the raw (c0) compression and model, the model demonstrates strong adaptability, achieving an 85% accuracy rate. However, the model's performance diminishes notably when exposed to LQ (c40) compression, dropping to a mere 20% accuracy or even a demolishing 0%. Nevertheless, this decline totally aligns with the findings in the FaceForensics++ paper, which highlighted lower accuracy in forgery detection for GAN-based methods like NeuralTextures. The unique model training approach of NeuralTextures, generating distinct models for each manipulation, introduces higher variation in possible artifacts, making the detection task even more challenging.

Original videos – Raw (c0) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000.mp4	0.99982	0.0001	Real	100%
003.mp4	0.9993	0.0007	Real	100%
012.mp4	0.9976	0.0024	Real	100%
015.mp4	0.9869	0.0131	Real	100%
024.mp4	0.9968	0.0032	Real	100%
026.mp4	0.999	0.001	Real	100%
029.mp4	0.9983	0.0017	Real	100%
035.mp4	0.9997	0.0003	Real	100%
036.mp4	0.9998	0.0002	Real	100%
044.mp4	0.999768	0.000232	Real	100%
047.mp4	0.996	0.004	Real	100%
048.mp4	0.9973	0.0027	Real	100%
073.mp4	0.99863	0.00137	Real	100%
078.mp4	0.9999	0.0001	Real	100%
102.mp4	0.9993	0.0007	Real	100%
114.mp4	0.9997	0.0003	Real	100%
128.mp4	0.99	0.01	Real	100%
135.mp4	0.9997	0.0003	Real	100%
138.mp4	0.999976	2.376e-05	Real	100%
141.mp4	0.9993	0.0007	Real	100%
Total Accuracy				**100.0%**

Table 7: Original videos – Raw (c0) compression – Raw (c0) model

Original videos – LQ (c40) compression – LQ (c40) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000.mp4	0.7994	0.2006	Real	100%
003.mp4	0.4804	0.5196	Real	**0%**
012.mp4	0.826	0.174	Real	100%
015.mp4	0.37	0.63	Real	**0%**
024.mp4	0.391	0.609	Real	**0%**
026.mp4	0.99	0.01	Real	100%
029.mp4	0.466	0.534	Real	**0%**
035.mp4	0.891	0.109	Real	100%
036.mp4	0.935	0.065	Real	100%
044.mp4	0.99996	4.424e-05	Real	100%
047.mp4	0.631	0.369	Real	100%
048.mp4	0.9996	0.0004	Real	100%
073.mp4	0.884	0.116	Real	100%
078.mp4	0.9993	0.0007	Real	100%
102.mp4	0.91	0.09	Real	100%
114.mp4	0.596	0.404	Real	100%
128.mp4	0.781	0.219	Real	100%
135.mp4	0.999963	3.514e-05	Real	100%
138.mp4	0.9992	0.0008	Real	100%
141.mp4	0.914	0.086	Real	100%
Total Accuracy				**80.0%**

Table 8: Original videos – LQ (c40) compression – LQ (c40) model

Original videos – LQ (c40) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000.mp4	0.998	0.002	Real	100%
003.mp4	0.987	0.013	Real	100%
012.mp4	0.92	0.08	Real	100%
015.mp4	0.944	0.056	Real	100%
024.mp4	0.992	0.008	Real	100%
026.mp4	0.931	0.069	Real	100%
029.mp4	0.985	0.015	Real	100%
035.mp4	0.989	0.011	Real	100%
036.mp4	0.998	0.002	Real	100%
044.mp4	0.997	0.003	Real	100%
047.mp4	0.936	0.064	Real	100%
048.mp4	0.979	0.021	Real	100%
073.mp4	0.985	0.015	Real	100%
078.mp4	0.9996	0.0004	Real	100%
102.mp4	0.994	0.006	Real	100%
114.mp4	0.996	0.004	Real	100%
128.mp4	0.794	0.206	Real	100%
135.mp4	0.991	0.009	Real	100%
138.mp4	0.989	0.011	Real	100%
141.mp4	0.939	0.061	Real	100%
Total Accuracy				**100.0%**

Table 9: Original videos – LQ (c40) compression – Raw (c0) model

The original videos, subjected to raw (c0) compression and model, consistently demonstrate remarkable accuracy, achieving almost a perfect detection rate. This brilliant performance can be attributed to the inherent nature of the original content, which serves as the ground truth for the model. The HQ raw model effectively discerns the authentic features present in the pristine videos, leading to precise and reliable classification. The success of the original videos in this configuration underscores the model's proficiency in distinguishing real content without the presence of distortions introduced by compression techniques. This performance stands in severe contrast to the challenges faced by manipulated videos, especially under LQ compression scenarios, where the model's accuracy experiences a notable decline. In addition, the 100% accuracy observed with LQ (c40) compression and Raw (c0) model configuration on original videos may be explained by the model consistently classifying these genuine videos as authentic. In contrast, other methods with the same configuration (Face2Face and NeuralTextures) achieve 0% accuracy, struggle to distinguish manipulated content under those settings.

In The Wild Fake videos – Raw (c0) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_unknown.mp4	0.9975	0.0025	Fake	0%
001_unknown.mp4	0.9632	0.0368	Fake	0%
002_unknown.mp4	0.9958	0.0042	Fake	0%
003_unknown.mp4	0.998	0.002	Fake	0%
004_unknown.mp4	0.9999929	7.143e-06	Fake	0%
005_unknown.mp4	0.998	0.002	Fake	0%
Total Accuracy				**0.0%**

Table 10: In The Wild Fake videos – Raw (c0) compression – Raw (c0) model

In The Wild Fake videos – LQ (c40) compression – LQ (c40) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_unknown.mp4	0.734	0.266	Fake	0%
001_unknown.mp4	0.9995	0.0005	Fake	0%
002_unknown.mp4	0.999907	9.265e-05	Fake	0%
003_unknown.mp4	0.9796	0.0204	Fake	0%
004_unknown.mp4	0.356	0.644	Fake	**100%**
005_unknown.mp4	0.988	0.012	Fake	0%
Total Accuracy				**0.167%**

Table 11: In The Wild Fake videos – LQ (c40) compression – LQ (c40) model

In The Wild Fake videos – LQ (c40) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000_unknown.mp4	0.862	0.138	Fake	0%
001_unknown.mp4	0.996	0.004	Fake	0%
002_unknown.mp4	0.9495	0.0505	Fake	0%
003_unknown.mp4	0.963	0.037	Fake	0%
004_unknown.mp4	0.9992	0.0008	Fake	0%
005_unknown.mp4	0.9981	0.0019	Fake	0%
Total Accuracy				**0.0%**

Table 12: In The Wild Fake videos – LQ (c40) compression – Raw (c0) model

These videos showcase a terrible outcome, as the model encounters difficulties distinguishing between real and manipulated content, resulting in an accuracy of 0%. This could be attributed to the external manipulation of these videos in diverse environments, potentially employing a variety of techniques beyond the scope of the Face2Face and NeuralTextures methods. Additionally, the inherent complexities introduced by real-world scenarios, such as instances where subjects may not be consistently front facing to the camera (as highlighted in the FaceForensics++ paper), the multitude of potential deepfake generation techniques, and variations in external manipulation methods contribute to the observed challenges. The model's struggle to handle such externally manipulated content underscores the limitations of these methods in addressing the broader spectrum of deepfake scenarios encountered in real-world settings.

Finally, the unusual result observed in both LQ (c40) compression and model, where only 1 out of 6 videos was classified correctly despite low probabilities (0.356% real vs. 0.644% fake), could be attributed to several factors. The weak probabilities suggest low confidence in the classification, and the limited success might be considered an outlier or a chance occurrence. Another potential explanation could be related to the specific characteristics of the videos. Since the video lacks dynamic facial movements or articulation, the model might struggle to clearly discern between manipulation and unaltered footage, resulting in correct predictions occurring by mere chance.

In The Wild Original videos – Raw (c0) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000.mp4	0.997	0.003	Real	100%
001.mp4	0.999	0.001	Real	100%
002.mp4	0.986	0.014	Real	100%
003.mp4	0.967	0.033	Real	100%
004.mp4	0.9795	0.0205	Real	100%
Total Accuracy				**100.0%**

Table 13: In The Wild Original videos – Raw (c0) compression – Raw (c0) model

In The Wild Original videos – LQ (c40) compression – LQ (c40) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000.mp4	0.397	0.603	Real	**0%**
001.mp4	0.997	0.003	Real	100%
002.mp4	0.941	0.059	Real	100%
003.mp4	0.616	0.384	Real	100%
004.mp4	0.568	0.432	Real	100%
Total Accuracy				**80.0%**

Table 14: In The Wild Original videos – LQ (c40) compression – LQ (c40) model

In The Wild Original videos – LQ (c40) compression – Raw (c0) model				
Video	**Average Real Probability**	**Average Fake Probability**	**Label**	**Accuracy**
000.mp4	0.997	0.003	Real	100%
001.mp4	0.985	0.015	Real	100%
002.mp4	0.662	0.338	Real	100%
003.mp4	0.614	0.386	Real	100%
004.mp4	0.426	0.574	Real	**0%**
Total Accuracy				**80.0%**

Table 15: In The Wild Original videos – LQ (c40) compression – Raw (c0) model

On the other hand, for these videos the raw (c0) compression with the raw (c0) model showcases impeccable accuracy, reaching 100%. This suggests the model's robustness in correctly identifying unaltered content in real-world scenarios. However, when introducing LQ (c40) compressions, the accuracy drops to 80%, indicating a slight decrease in performance. This decline could be attributed to the lower quality of compressed data, impacting the model's ability to precisely distinguish authentic content, though still maintaining a respectable accuracy level.

These results emphasize the importance of considering real-world, externally manipulated videos separately from those generated by specific methods. The intricacies and variations introduced by external manipulation can significantly influence the performance of deepfake detection models.

Additionally, analyzing the results globally, the observed decline in performance when utilizing LQ compression with HQ models across various methods can be attributed to a misalignment in the available features and information for the models. LQ compression introduces substantial information loss and distortion, degrading overall image or video quality. The mismatch between the expected detail level by the HQ model and the reduced information in LQ compressed data prevents effective generalization, resulting in diminished accuracy.

When examining the results collectively, a pattern emerges highlighting the critical role of raw data quality. Both the Face2Face and NeuralTextures methods exhibit superior performance when operating on raw quality videos. The vulnerability of high-quality models to lower-quality compressions is obvious, emphasizing the need for careful consideration in real-world deployment where data quality variations are inevitable. The challenges encountered with in-the-wild fake videos underscore the complexities of detecting deepfakes in dynamic, unpredictable settings. In conclusion, these insights demonstrate the importance of continuous research and advancements to enhance model adaptability and robustness in addressing real-world challenges.

Finally, the processing times for each method under various configurations were recorded and are presented in the chart below, offering a snapshot of the time efficiency associated with each method in different scenarios, providing a glimpse into the computational demands it implies.

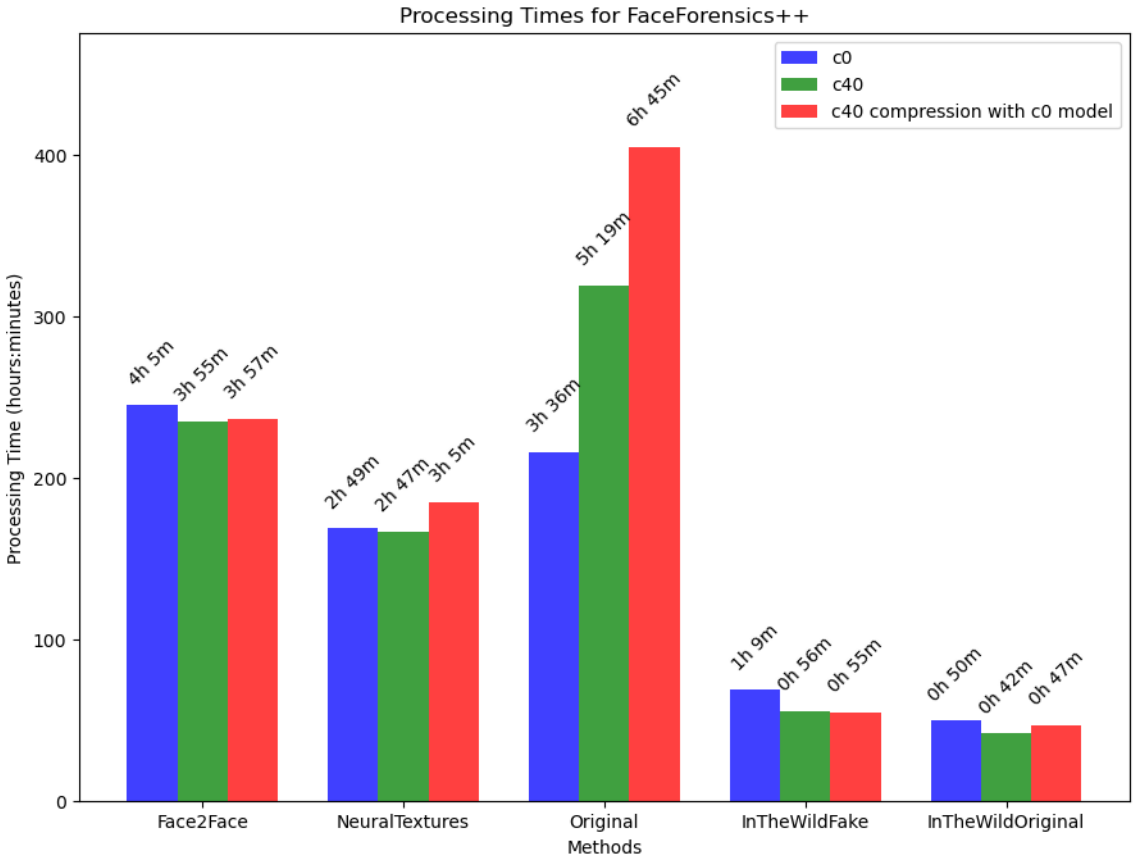

Figure 6: Accuracy Processing Times for FaceForensics++ dataset. Tested with an Apple M2 Max Chip / 32GB RAM / 30-core GPU / 12-core CPU

2.2 LipForensics

2.2.1 Overview

The next face forgery detection algorithm tested was LipForensics [14] (also publicly available on GitHub [15]). It introduces a novel approach addressing the limitations of current deep learning-based detectors. While existing detectors excel in constrained scenarios, they struggle with samples created by unseen manipulation methods, particularly when subjected to common post-processing operations like compression. LipForensics aims to overcome these challenges by focusing on high-level semantic irregularities in mouth movements, a common trait in many generated videos. It distinguishes itself by capitalizing on the observation that most face video forgeries alter the mouth in some way to match it with someone else's identity, speech, or expression. The complicated motion of the mouth poses a challenge for manipulation methods, leading to difficulties in generating movements without falling into the "uncanny valley" (unease experienced by humans when observing a realistic computer-generated face).

As a result, the method employs a two-step approach to achieve this goal. First, a convolutional neural network (CNN) is pretrained, consisting of a spatiotemporal feature extractor followed by a temporal convolutional network, specifically on the task of lipreading. This process produces internal representations to be sensitive to anomalous dynamics of the mouth in a high-level semantic space. The reasoning behind this lies in recent anomaly detection literature, which suggests that training on the "normal" class (real videos) for a relevant task promotes learning features useful for detecting "anomalous" samples (fake videos).

In the second step, the feature extractor is frozen, and only the temporal network is finetuned on forgery data. This strategic choice prevents the network from relying on unwanted artifacts instead of focusing on mouth movements. Unlike other approaches that address overfitting by blurring or adding noise to the input, LipForensics passes video clips through a deep feature extractor pretrained to perform lipreading. In this way, output embeddings exhibit relative invariance to low-level artifacts, ensuring a robust representation.

The finetuning phase involves feeding 25 grayscale, aligned mouth crops through the frozen feature extractor, initially implemented as a ResNet-18 with an initial 3-D convolutional layer. This feature extractor, pretrained on lipreading, outputs embeddings sensitive to mouth movements. Subsequently, a multi-scale temporal convolutional network (MS-TCN), also pretrained on lipreading, is finetuned to detect fake videos based on semantically high-level irregularities in mouth motion.

In terms of preprocessing, RetinaFace [16][17] is employed for face detection in each frame, extracting the largest face. In LipForensics, mouths are cropped by computing 68 facial landmarks with the Face Alignment Network (FAN) [18][19]. These landmarks are smoothed over 12 frames to mitigate motion jitter, and each frame undergoes affine warping to the mean face using five landmarks around the eyes and nose. Mouth cropping involves resizing the image and extracting a fixed region centered around the mean mouth landmark. Additionally, alignment is performed to eliminate translation, scale, and rotation variations without altering the natural movement of the mouth.

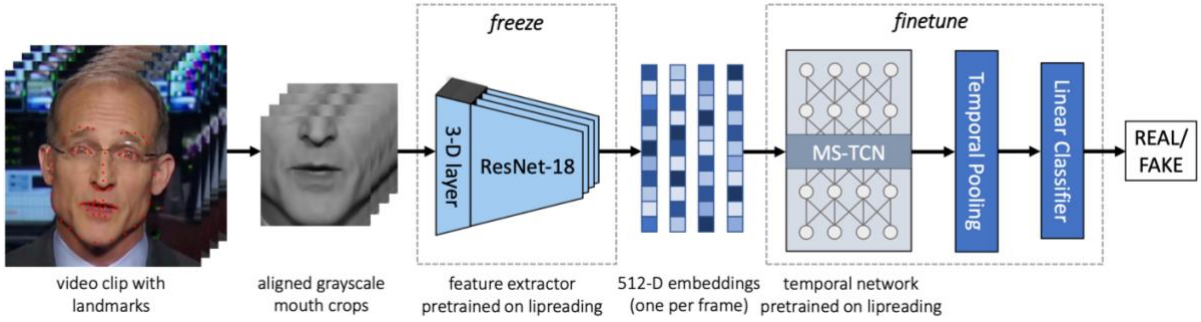

Figure 7: Overview of LipForensics' workflow (diagram taken from the authors' paper)

Moreover, LipForensics leverages various datasets for comprehensive training and evaluation. The primary dataset is FaceForensics++, but it also utilizes DeeperForensics and FaceShifter, both featuring improved face swapping algorithms applied to real videos from FF++. Additionally, the test set of Celeb-DF-v2, a face-swapping dataset, and 3,215 test set videos from the DeepFake Detection Challenge (DFDC) contribute to the evaluation, offering diverse scenarios with extreme conditions such as large poses and low lighting.

Regarding the metrics, results are reported using accuracy area under the receiver operating characteristic curve (AUC). To ensure fair comparison with models using a single frame as input, LipForensics computes video-level measures by averaging model predictions across the entire video, aligning with established practices in the literature.

Furthermore, the extensive experiments conducted demonstrate the effectiveness of LipForensics in terms of generalization to unseen manipulations and robustness to perturbations. The approach significantly outperforms previous methods, showcasing its superiority in various challenging scenarios. In addition, the method proves to be effective even on heavily compressed data, a feature that sets it apart from other detectors. Finally, LipForensics contributes a valuable perspective to the field, emphasizing the importance of semantically high-level cues in building a reliable face forgery classifier.

Despite its strong performance in various experiments, LipForensics has limitations. It is not applicable to isolated images and might not effectively detect fake videos where the mouth is occluded or remains unaltered. While the model excels in identifying manipulations involving speech, identity, or expression, there may be a performance decline in scenarios with limited mouth motion. Additionally, the model relies on a large-scale labeled dataset for effective pretraining.

2.2.2 Performed Work

Once again, following a comprehensive description of the mechanisms and principles of the LipForensics method, this section outlines the principal activities conducted on the dataset. It captures the activities undertaken, the accomplished results, the decisions made, and any challenges encountered.

The primary objective was to replicate a scaled-down version of the LipForensics dataset, employing the 20 videos previously utilized in FaceForensics++ as the test set. This approach allowed for a comparative analysis between both algorithms, facilitating the reach of conclusions and insights, in addition to evaluating the efficacy of the new method.

The initial step, as clearly exposed by the algorithm, involved acquiring the videos along with their corresponding images or frames. This process was almost instantaneous, since we already had extracted the 20 videos from the test set (considering the use of the same two quality-level compressions). However, the focus now shifted to extracting the frames from each video, therefore enabling the isolation of the important region in the image, which is the mouth area, as detailed earlier.

To accelerate the frame's extraction process, we employed a Python script ("extract_compressed_videos"). The functionality of this script centered on extracting images from a designated folder, significantly aiding in the extraction of a substantial number of frames per video and per method (including Face2Face, NeuralTextures, original videos, and in-the-wild videos).

Figure 8: First frame of the second video (003_000) from the Face2Face videos test set (using raw quality)

After completing this procedure, the following step involved extracting 68 landmarks from the recently acquired frames of the videos. To achieve this, the initial requirement was to detect the face (or faces, considering that many videos contain more than one face) in each image. Initially, as detailed in the paper, the RetinaFace algorithm was considered for this purpose, aiming to detect every face in each frame of the videos. Yet, while working on this phase for some time, it became obvious that this approach was excessively time-consuming.

The main challenge of the LipForensics algorithm emerged from the considerable demand of processing time. This was primarily associated with the need of first detecting the primary face in each frame and subsequently extracting the landmarks from that face (and discarding the rest) for later cropping the mouth region associated, which is the crucial step for the LipForensics algorithm to operate effectively. Consequently, we opted for an alternative approach. Instead of first using RetinaFace to detect faces and later extracting landmarks from the main face out of all the faces detected with the Face Alignment Network (which was the initial plan), we modified the FAN algorithm's code. As a result, this modification allowed the extraction of the primary or main face from all detected faces by discarding smaller bounding boxes and retaining only the largest bounding box, as opposed to the previous workflow, which implied extracting the landmarks of each face detected instead of the main one.

Figure 9: Example (frame 19 of video 048 from the raw quality original sequences) showcasing how RetinaFace detects faces

This decision significantly sped this process up, reducing processing time and resources. Nevertheless, given that each video comprised between 300 (for the shortest ones) and 800 (for the longest ones) frames, extracting landmarks for an entire video remained time demanding, considering the massive number of videos the dataset possessed. To address this, my supervisor and I made the decision to further reduce the test set, going from the initial 20 videos to just applying the first 10. While being aware of the potential negative impact on results that this decision meant, the time constraints imposed on our project demanded this adjustment.

Having optimized the process for detecting the primary face in each frame, we developed a straightforward Python script to extract the 68 landmarks using the modified FAN algorithm. Then, the next phase involved cropping the mouth region from the mentioned frames. To achieve this, we utilized the "crop_mouths.py" Python script from the LipForensics code, which essentially extracts the aligned cropped mouths from the images of the test set videos. Still, adjustments were made to the LipForensics code, including the modification of the "crop_mouths.py" file, so the new dataset used in FF++ (the in-the-wild videos) was accommodated. Once the code was updated accordingly, the cropped mouths for each method (F2F, NT, original videos, in-the-wild real and fake videos) and each compression (raw and c40) were successfully obtained.

Figure 10: Example (frame 28 of video 005_unknown from the raw quality in-the-wild fake videos) displaying the frame (left) and its corresponding cropped mouth region (right)

Finally, to evaluate the robustness and accuracy of this algorithm, the author's Python script "evaluate.py" was executed (considering that the in-the-wild adjustments had to be applied, as with the "crop_mouths.py" code). This script serves the purpose of assessing the performance of the LipForensics model across diverse face forgery datasets. The evaluation metric employed is the Area Under the ROC Curve (AUC) at the video level, offering a complete view of model effectiveness since it is particularly well-suited for binary classification problems, such as distinguishing between genuine and manipulated videos, which is the essence of forgery detection in the LipForensics model. This systematic approach not only ensures reproducibility but also facilitates easy comparison of results across different datasets, contributing to a robust evaluation of the LipForensics model.

2.2.3 Results

This section offers a comprehensive exploration into the performance of face forgery detection methods across various compressions and models. Through meticulous evaluation, we explore the details of the distinct methods seen and the analysis of manipulated, original and "in the wild" videos. By examining the outcomes under different conditions, we aim to discern patterns and reveal key insights about these algorithms.

As mentioned earlier, a primary challenge encountered in this project was efficiently managing the extraction of such a considerable number of landmarks within the limited timeframe. To illustrate the processing times we had to face, the following bar chart provides a visual representation of the tremendous time required to process the landmarks from each method and compression (considering that there are 10 videos in F2F, NT and original sequences, 6 videos in InTheWildFake and 5 videos in InTheWildOriginal).

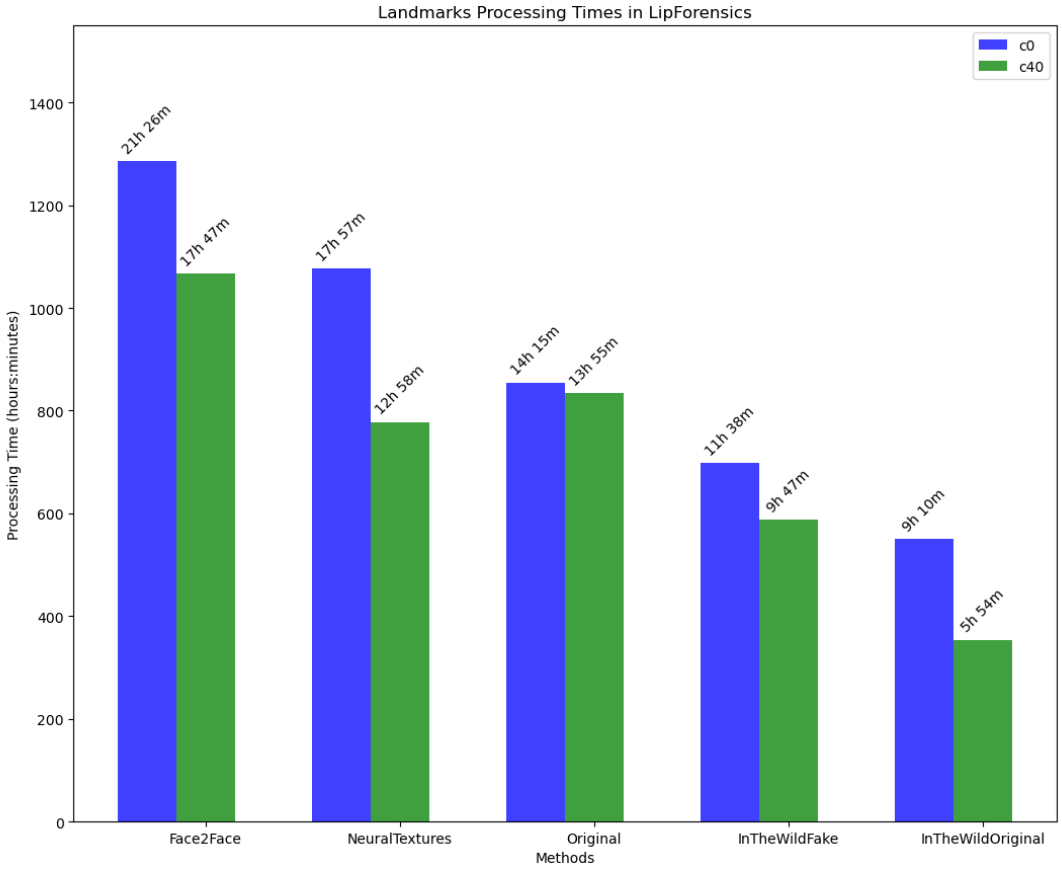

Figure 11: Landmarks Processing Times for LipForensics dataset. Tested with an Apple M2 Max Chip / 32GB RAM / 30-core GPU / 12-core CPU

The presented results are obtained by executing the "evaluation.py" Python script, as previously described. However, it must be remarked that in the assessment of face forgery datasets, metrics for the original sequences are not included. This is because the primary objective in forgery detection is to identify manipulated or fake content, consequently, evaluating the model's performance on datasets that exclusively consist of manipulated videos aligns with the core goal of the forgery detection task, enabling the evaluation to become more pertinent to the specific objectives at hand.

Dataset	**Compression**	**Area Under the ROC Curve (at video-level)**
Face2Face	Raw (c0)	1.0%
Face2Face	Low-quality (c40)	0.87%
NeuralTextures	Raw (c0)	1.0%
NeuralTextures	Low-quality (c40)	0.81%
InTheWildFake	Raw (c0)	0.5%
InTheWildFake	Low-quality (c40)	0.57%

Table 16: Results for LipForensics dataset

These results reveal notable challenges in the model's ability to distinguish between real and manipulated content, resulting in accuracies ranging from 0.5% to 1.0% (out of 1.0%) across different scenarios, which at first glance demonstrates worse performance than the FaceForensics++ algorithm.

In the case of Face2Face, the model achieves a perfect score (1.0%) under raw (c0) compression, indicating a strong ability to identify manipulated videos. However, when subjected to LQ (c40) compression, the accuracy decreases to 0.87%, suggesting increased difficulty in discerning manipulated content. A similar pattern is observed in NeuralTextures, where the model performs well under raw (c0) compression (1.0%) but faces challenges under LQ (c40) compression (0.81%).

The observed decline in accuracy for both methods implies that the model encounters challenges in sustaining its forgery detection capabilities under aggressive video compression. This outcome aligns with the findings in the FaceForensics++ dataset, indicating a consistent behavior across datasets. It underscores the importance for model robustness in real-world scenarios characterized by diverse video qualities. The impact of compression-induced information loss, artifacts, and heightened noise becomes evident in the diminished performance, highlighting the importance of developing models that can effectively navigate the complexities introduced by various compression scenarios.

In the context of InTheWildFake, the model encounters substantial difficulties, achieving accuracies of 0.5% and 0.57% under raw (c0) and LQ (c40) compressions, respectively. These lower accuracies (compared to the FF++ ones) may be attributed to the inherent challenges presented by real-world scenarios, since these videos are captured in diverse and uncontrolled environments, introducing complexities such as varying lighting conditions, camera angles, and background settings. These factors pose difficulties for the model, likely introducing variations that make it more challenging for the model to discern between genuine and manipulated content accurately.

Furthermore, a singular occurrence was observed in the case of video 002 within the InTheWildOriginal dataset. Upon comparing the results of mouth region cropping between the two compression settings, a discrepancy became evident. In the raw video quality, the algorithm encountered difficulties in accurately extracting the mouth region, in contrast to the c40 compression where the extraction performed relatively better. These disparities may be attributed to the specific characteristics and settings of this particular video. Notably, in the raw quality video, the algorithm erroneously identified the microphone as a face, leading to inaccurate mouth region extraction. Conversely, in the c40 compression, such misidentification did not occur, highlighting a distinct improvement and hence resulting in a higher accuracy of 0.57%, compared to 0.5% in the raw video quality.

Additionally, it must also be considered the fact that, instead of testing on 20 videos (as in FF++), this method was only tested on 10 videos, reducing the test size in half. Again, while this reduction is suboptimal, it was a necessary decision due to the time constraints.

Figure 12: Theorical "faces" detected on frame 21 of video 002 from InTheWildOriginal videos

Finally, it must be highlighted a notable limitation of LipForensics, as previously discussed: its sensitivity to occlusion. The algorithm operates under the assumption that the mouth region is visibly present in an image. Consequently, when the mouth is not clearly detectable, the performance of this algorithm significantly drops. This limitation is exemplified below with these three consecutive frames (272, 273, and 274) from video 002 within the InTheWildOriginal dataset. Specifically, in frame 273, which corresponds to a transition in the original YouTube video, the mouth region is occluded. Consequently, the algorithm fails to extract landmarks from this frame, resulting in a suboptimal performance during the assessment of its capabilities. This emphasizes the algorithm's vulnerability to scenarios where the mouth region is obscured or not clearly visible.

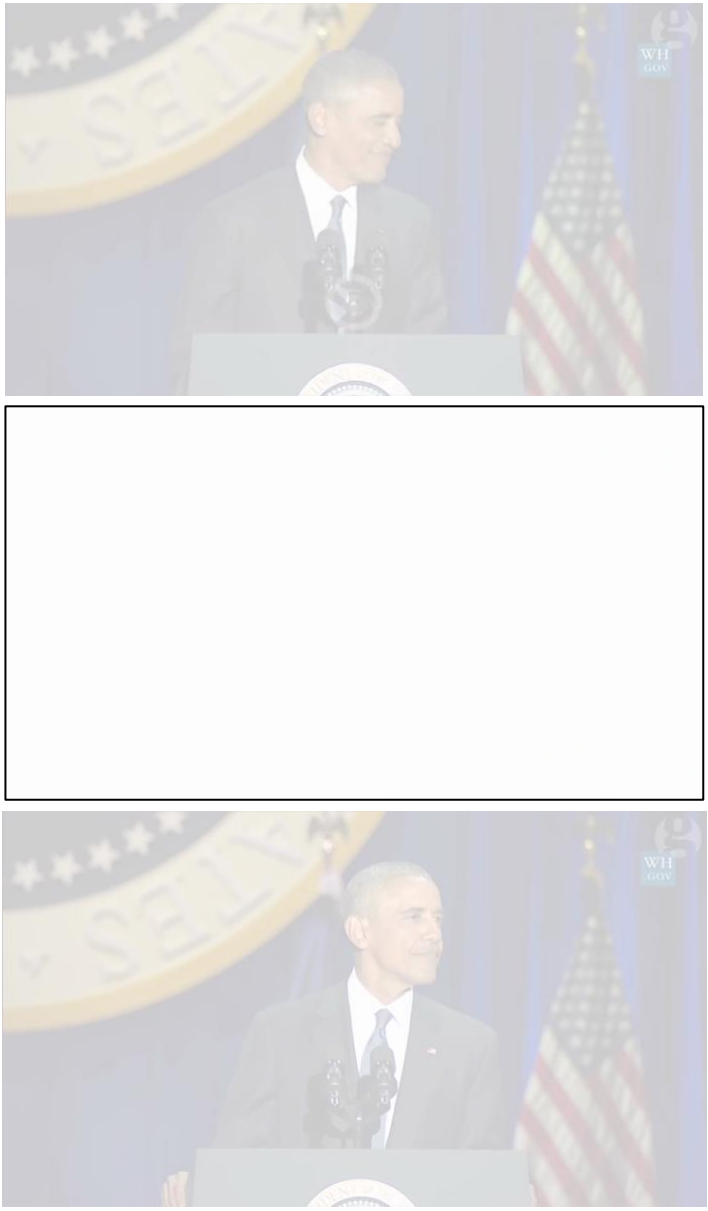

Figure 13: Frames 272 (upper), 273 (middle, outlined so it is more visible) and 274 (lower) of video 002 from InTheWildOriginal dataset

2.3 ID-Reveal and Person-of-Interest

2.3.1 Overview

2.3.1.1 ID-Reveal

The last section of this chapter centers on two closely related methods: ID-Reveal [20] and Person-of-Interest (POI) Forensics [21]. The decision to analyze both methods together roots from the recognition that understanding ID-Reveal is crucial for a comprehensive assessment of POI-Forensics, because the authors from both works are almost the same people, explicitly acknowledging that the main code of ID-Reveal [22] is contained in the GitHub repository of POI-Forensics [23].

ID-Reveal introduces a novel strategy centered on learning temporal facial features associated with a person's unique speaking patterns. This is achieved through metric learning coupled with an adversarial training strategy, eliminating the need for training data of fake videos, and relying solely on real videos. The incorporation of high-level semantic features enhances robustness to various post-processing techniques, contributing to improved generalization across different facial manipulations.

The architecture of ID-Reveal comprises three key components: a facial feature extractor utilizing a 3D Morphable Model (3DMM), a temporal network for detecting biometric anomalies (temporal ID network), and a generative adversarial network (GAN) responsible for predicting person-specific motion based on expressions from a different subject. Notably, the model is trained exclusively on real videos featuring diverse subjects, promoting adaptability to different manipulation methods encountered in real-world scenarios.

The method's evaluation demonstrates its efficacy in generalizing across various manipulation types, even in the presence of low-quality videos, achieving a significant average improvement of over 15% compared to state-of-the-art approaches (bearing in mind that this paper/work was released in August 2021). ID-Reveal shifts the focus from a binary classification of real or fake to a more nuanced assessment, aiming to reveal whether the face under examination preserves all biometric traits of the involved subject.

The process begins with the extraction of facial features using the mentioned 3D Morphable Model (3DMM). This model captures a low-dimensional representation of a face based on a combination of principal components that describe shape, expression, and appearance. The 3DMM is trained to predict a vector of 62 coefficients for each frame, representing the necessary facial parameters. This feature extraction is crucial as it provides a consistent and detailed representation of the facial characteristics, which are then used for further analysis.

The extracted facial features are then fed into the Temporal ID Network, a crucial component of the ID-Reveal architecture. This network processes the temporal sequence of 3DMM features through convolutional layers that operate along the temporal dimension. The purpose of this network is to compute an embedded vector that captures the temporal dynamics of facial features. By evaluating the distance between these embedded vectors, the model can assess the consistency of facial movements over time. The Temporal ID Network is trained to maximize the similarity of these vectors for the same individual across different frames, while minimizing the similarity for different individuals. Consequently, this metric learning approach ensures that the model focuses on the unique temporal patterns of each person's facial movements, thus enhancing its ability to detect anomalies indicative of manipulation.

To further refine the detection capabilities, ID-Reveal incorporates a Generative Adversarial Network (GAN). This GAN consists of two components: the Temporal ID Network, which acts as the discriminator, and the 3DMM Generative Network, which functions as the generator. The GAN is trained in an adversarial manner, where the GAN attempts to produce realistic sequences of facial features that can fool the Temporal ID Network. The discriminator, in turn, learns to distinguish between real and generated sequences. This adversarial training encourages the Temporal ID Network to focus on temporal aspects rather than visual cues alone, making it more robust to various types of facial manipulations.

The training process for ID-Reveal is exclusively based on real videos, which contain a diverse set of subjects and scenarios. By avoiding the use of fake videos during training, the model learns to detect anomalies based on the inherent characteristics of genuine facial movements, rather than specific artifacts introduced by known manipulation techniques. This approach significantly improves the generalization capabilities of the model, allowing it to detect previously unseen types of forgeries effectively.

2.3.1.2 Person-of-Interest Forensics

On the other hand, the POI-Forensics introduces a person-of-interest (POI) deepfake detector grounded in the belief that each individual possesses unique characteristics that synthetic generators struggle to replicate. The approach involves extracting audio-visual features characterizing a person's identity and utilizing them to create an effective deepfake detection system. This system focuses on leveraging intrinsic identity characteristics, which synthetic generators cannot accurately reproduce, thereby facilitating the detection of deepfakes through inconsistencies in these features.

A distinctive feature of POI-Forensics is its adoption of a contrastive learning paradigm, leveraging a multi-modal analysis that encompasses both audio and video cues. The training process involves learning embeddings for face and audio segments, ensuring discriminative representations for each identity. Specifically, the model learns to map segments of the same identity close to one another in the embedding space, while mapping segments from different identities far apart. This enables the detector to handle both single-modality (audio-only, video-only) and multi-modality (audio-video) attacks, showcasing flexibility in addressing various manipulation scenarios.

To achieve this, the model employs a contrastive loss function during training, which ensures that the learned embeddings for the same person are more similar than those for different people. This loss function drives the network to minimize the distance between embeddings of the same identity and maximize the distance between embeddings of different identities (similar to ID-Reveal). The model utilizes both audio and visual features extracted from the input data. The visual features are derived from a face recognition network, while the audio features are extracted using an audio recognition network (both neural networks are based on ResNet-50 with Group-Normalization). These features are then fused to create a comprehensive representation of the person's identity.

The multi-modal analysis is particularly powerful because it allows the system to capture the synchronized patterns of a person's facial movements and speech characteristics. This synchronization is difficult to fake convincingly, providing a robust basis for detecting deepfakes. The model's architecture includes separate branches for processing audio and video inputs, which are later combined in a joint embedding space. This design enables the model to learn complex correlations between audio and visual data, enhancing its ability to detect inconsistencies that indicate manipulation.

The method's key contributions include its capability for generalization, robustness, and flexibility. Notably, training exclusively on real talking-face videos ensures good generalization to known and unknown manipulations, irrespective of the manipulation method employed. The model's flexibility is evident in its ability to detect video-only, speech-only manipulations, and even scenarios where a real audio track is swapped onto the original video. The robustness of the detector is demonstrated in its resilience to challenging real-world conditions, such as compressed or maliciously modified videos.

To further emphasize its effectiveness, POI-Forensics surpasses state-of-the-art (considering the paper/work was released in May 2023) approaches by a significant margin, particularly excelling in scenarios involving compressed videos. The method's success lies in its ability to exploit audio-visual features to reveal anomalies, steering away from traditional approaches focused on detecting generation artifacts. By training exclusively on real videos and learning a real-world data model, POI-Forensics demonstrates a high level of independence from specific manipulation methods, underlining its intrinsic high generalization ability.

2.3.2 Performed Work

As extensively discussed in this paper, the primary constraining factor throughout this research has been the limitation of time. Unfortunately, due to this time constraint, we found ourselves with limited opportunities to thoroughly evaluate these last detection methods. As the project timeline approached its conclusion, we faced computational and processing challenges in conducting the testing for these methods, particularly considering that they are complex algorithms with both audio-visual features.

Furthermore, during our in-depth investigation of the code associated with these methods, my supervisor and I discovered that the provided code was incomplete. Specifically, the model's weights were missing, which significantly hindered our ability to conduct comprehensive evaluations. Although the GitHub repository included an example and the weights related to it, running the model independently proved to be extremely time-consuming, taking approximately five hours for just a single iteration. This limitation severely restricted our capacity to perform extensive testing.

Consequently, the evaluation of the methods was shortened, only being able to test 2 videos, video "000_unknown" and video "003" from the in-the-wild fake and original videos respectively. The obtained results are presented below, which, given the potential of these algorithms, we believe that they could exhibit significantly enhanced performance under more extensive testing scenarios. Testing the algorithm with only one 15-second video cannot fully showcase its potential performance, therefore, while the preliminary results are insightful, they should be viewed as a starting point for future, more exhaustive investigations into the capabilities and limitations of these cutting-edge detection methods.

2.3.3 Results

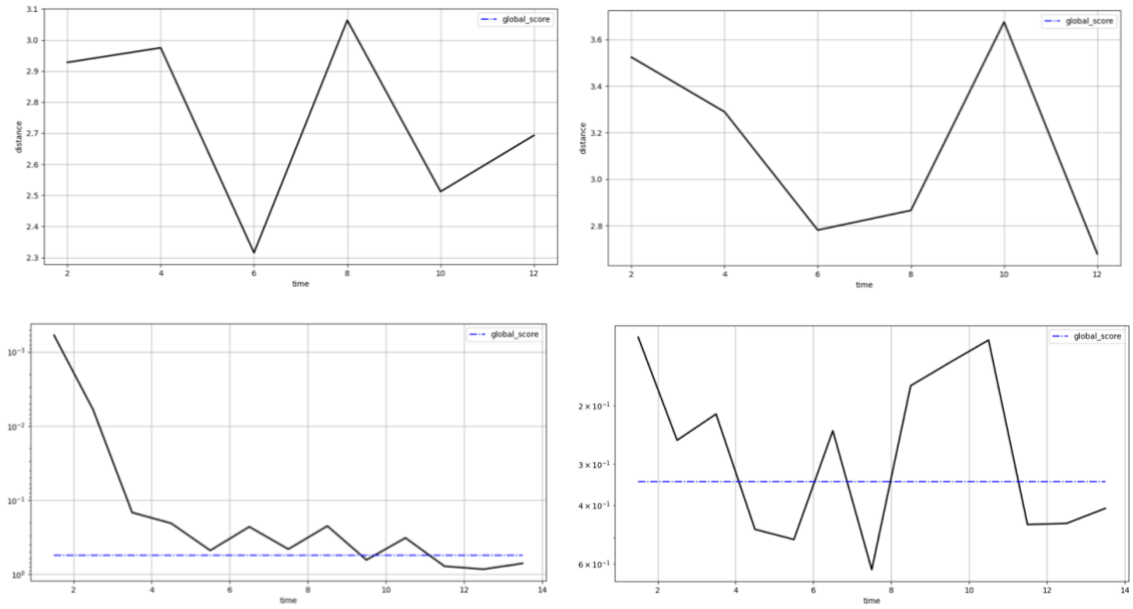

Figure 14: Results from ID-Reveal and POI-Forensics. Upper graphs are real (left) and fake (right) Boris Johnson with ID-Reveal while lower graphs are the same but for POI-Forensics

In the following analysis, we present four distinct graphs, each associated with videos featuring the UK ex-prime minister Boris Johnson. The upper graphs showcase results obtained from the ID-Reveal method. The left graph corresponds to video 003, depicting the real Boris Johnson in in-the-wild original videos, while the right graph relates to video 000_unknown, portraying a manipulated or fake Boris Johnson. Moving to the lower graphs, these belong to the POI-Forensics method. The left graph represents the genuine video, and the right one depicts the manipulated or fake video, aligning with the videos mentioned above.

Notably, the results from POI-Forensics surpass those from ID-Reveal. One significant distinction is the availability of a global score in POI-Forensics, approximately reaching 0.3. This score suggests higher similarity between the videos, as a higher number corresponds to more manipulation. Examining the upper graphs, numerous peaks or abrupt increases in distances are evident, indicating potential points in the video where manipulation or tampering has occurred. In contrast, the lower graphs reveal that the genuine video maintains consistently low distances, indicative of a video without manipulation.

In conclusion, POI-Forensics demonstrates superior capabilities in distinguishing between genuine and manipulated videos compared to the ID-Reveal method. However, it's crucial to note that these results fall short of the desired optimal performance. Ideally, real videos should yield a global score of 0 and display minimal peaks in the distance graphs, while manipulated ones should register a higher global score with noticeable peaks. Unfortunately, due to time constraints limiting the testing of these methods with more pristine videos, the observed results are suboptimal.

3 Results and conclusions

This work aimed to conduct a comprehensive exploration into the realm of deepfake detection, revealing critical insights into the complexities and challenges associated with identifying manipulated content. The focus was particularly on the FaceForensics++, LipForensics, ID-Reveal and POI-Forensics methods. Deepfake detection serves as an important defense against the growing threats posed by synthetic media, as the ability to discern between authentic and manipulated content is crucial in preventing potential consequences such as misinformation, privacy breaches, and the erosion of trust in digital media.

Our investigation covered a diverse array of datasets, each presenting unique challenges and characteristics. The methods analyzed demonstrated promising capabilities in terms of generalization and robustness, addressing some limitations observed in traditional detection methods. The strengths and weaknesses identified contribute to a better understanding of their applicability in real-world scenarios.

FaceForensics++ dataset provided a robust foundation for testing detection algorithms targeting facial reenactment techniques such as Face2Face and NeuralTextures. Despite its novel approach, the performance of FaceForensics++ was inconsistent, especially when tested on compressed or in-the-wild videos.

LipForensics introduced an innovative method focused on the semantic consistency of lip movements, aimed at detecting deepfakes generated through lip-synching. The experiments demonstrated that models trained on this dataset could detect manipulations effectively, particularly when the mouth region was emphasized. However, the performance varied depending on video quality and compression levels, underscoring the importance of considering compression settings and occlusion sensitivity.

ID-Reveal method focused on the analysis of identity consistency across frames in videos, leveraging identity verification techniques to detect inconsistencies that may arise in deepfakes. Results indicated that ID-Reveal was relatively effective in maintaining identity integrity, particularly in cases where face-swapping techniques were used. However, the method showed limitations when applied to videos with high levels of noise or compression, where identity features could be blurred or distorted.

Person-of-Interest (POI) method aimed at detecting deepfakes by analyzing the behavioral and appearance consistency of a specific individual across different video segments. This method was particularly useful in scenarios involving well-known public figures. The evaluation revealed that POI was effective in identifying discrepancies in behavior or appearance that are typical in deepfakes, especially in controlled settings. However, its performance diminished when applied to lower-quality or highly compressed videos.

FaceForensics++ and LipForensics highlighted the importance of considering compression settings and occlusion sensitivity, while ID-Reveal and POI-Forensics showcased advancements in generalization and multi-modal detection.

Despite valuable insights gained from our analysis, temporal and computational constraints imposed limitations on the extent of our evaluations. The downsized test set emphasizes the need for future attempts to encompass a more comprehensive evaluation with the complete set of 140 test videos. Looking ahead, future works should prioritize an in-depth exploration of the remaining test set videos to refine the evaluation and enhance the overall efficacy of the methods.

In conclusion, this work underscores the ever-evolving landscape of deepfake detection, emphasizing the continual need for advancements to stay ahead of challenges posed by rapidly advancing face manipulation technologies. The insights gained here serve as a foundation for ongoing efforts to fortify digital media integrity and combat the spread of fake content in the digital age.

4 Bibliography

[1] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, Matthias Nießner. (2019). *FaceForensics++: Learning to Detect Manipulated Facial Images.* [Online]. Available: https://arxiv.org/pdf/1901.08971.pdf

[2] ondyari. (2019). *FaceForensics.* [Online]. Available: https://github.com/ondyari/FaceForensics

[3] YouTube. (2023, October 20th). *FULL SPEECH: President Biden delivers address to the nation | ABC News.* [Online]. Available: https://www.youtube.com/watch?v=b1ukaC9cSKQ&t=296s

[4] YouTube. (2019, October 30th). *Speech by Angela Merkel, Chancellor of Germany (DE).* [Online]. Available: https://www.youtube.com/watch?v=cDEwenlY9Xo&t=97s

[5] YouTube. (2017, January 11th). *Barack Obama's final speech as president – video highlights.* [Online]. Available: https://www.youtube.com/watch?v=k0jJL_YFyIU&t=2s

[6] YouTube. (2022, September 6th). *Boris Johnson's farewell speech as UK prime minister – BBC News.* [Online]. Available: https://www.youtube.com/watch?v=9Gf5X1BC7tY&t=110s

[7] YouTube. (2015, June 17th). *Donald Trump's best lines during his 2016 speech.* [Online]. Available: https://www.youtube.com/watch?v=f0UB06v7yLY&t=14s

[8] YouTube. (2019, November 26th). *Boris Johnson has a message for you. (deepfake).* [Online]. Available: https://www.youtube.com/watch?v=30NvDC1zcL8

[9] YouTube. (2021, July 7th). *This is not Morgan Freeman - A Deepfake Singularity.* [Online]. Available: https://www.youtube.com/watch?v=oxXpB9pSETo

[10] YouTube. (2023, May 15th). *Now Bill Gates...* [Online]. Available: https://www.youtube.com/watch?v=_LxFA2p_gmY

[11] YouTube. (2018, April 17th). *You Won't Believe What Obama Says In This Video!* [Online]. Available: https://www.youtube.com/watch?v=cQ54GDm1eL0&t=1s

[12] YouTube. (2020, September 29th). *Dictators - Vladimir Putin.* [Online]. Available: https://www.youtube.com/watch?v=sbFHhpYU15w

[13] YouTube. (2022, November 29th). *Fake Zuck to Congress: Thank You for Inaction on Big Tech Antitrust!* [Online]. Available: https://www.youtube.com/watch?v=5Fv-LKT_cEc&t=28s

[14] Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, Maja Pantic. (2021). *Lips Don't Lie: A Generalisable and Robust Approach to Face Forgery Detection.* [Online]. Available: https://arxiv.org/pdf/2012.07657.pdf

[15] ahaliassos. (2021). *LipForensics.* [Online]. Available: https://github.com/ahaliassos/LipForensics

[16] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, Stefanos Zafeiriou. (2019). *RetinaFace: Single-stage Dense Face Localisation in the Wild.* [Online]. Available: https://arxiv.org/pdf/1905.00641.pdf

[17] biubug6. (2021). *Pytorch_Retinaface.* [Online]. Available: https://github.com/biubug6/Pytorch_Retinaface

[18] Adrian Bulat, Georgios Tzimiropoulos. (2017). *How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks).* [Online]. Available: https://openaccess.thecvf.com/content_ICCV_2017/papers/Bulat_How_Far_Are_ICCV_2017_paper.pdf

[19] 1adrianb. (2021). *face-alignment.* [Online]. Available: https://github.com/1adrianb/face-alignment

[20] Davide Cozzolino, Andreas Rössler, Justus Thies, Matthias Nießner, Luisa Verdoliva. (2021). *ID-Reveal: Identity-aware DeepFake Video Detection.* [Online]. Available: https://arxiv.org/pdf/2012.02512.pdf

[21] Davide Cozzolino, Alessandro Pianese, Matthias Nießner, Luisa Verdoliva. (2023). *Audio-Visual Person-of-Interest DeepFake Detection.* [Online]. Available: https://arxiv.org/pdf/2204.03083.pdf

[22] grip-unina. (2021). *id-reveal.* [Online]. Available: https://github.com/grip-unina/id-reveal

[23] grip-unina. (2023). *poi-forensics.* [Online]. Available: https://github.com/grip-unina/poi-forensics