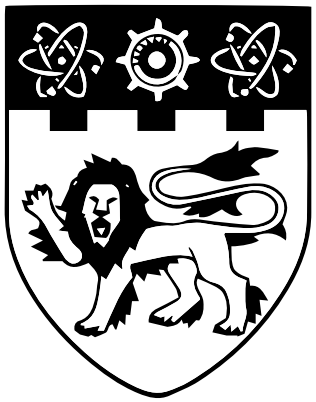

NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

NO MATTER SMALL OR BIG LIP MOTION: DEEPFAKE DETECTION WITH REGULARIZED FEATURE LEARNING ON SEMANTIC INFORMATION

YANG ZHIYUAN

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

2024

NO MATTER SMALL OR BIG LIP MOTION: DEEPFAKE DETECTION WITH REGULARIZED FEATURE LEARNING ON SEMANTIC INFORMATION

YANG ZHIYUAN

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Master of Engineering

2024

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

15/08/2023
. . . . . .
Date

. . . . . .
Zhiyuan Yang

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

17/08/2023

. . . . . .

Date

. . . . . .

Bihan Wen

Authorship Attribution Statement

This thesis contains material from 11 paper(s) published in the following peer-reviewed papers accepted at conferences in which I am listed as an author.

Chapter 3 and 4 are accepted as Z. Yang, et al. "No Matter Small or Big Lip Motion: DeepFake Detection with Regularized Feature Learning on Semantic Information." 2023 IEEE 6th International Conference on Multimedia Information Processing and Retrieval (MIPR). IEEE Computer Society, 2023.

The contributions of the co-authors are as follows:

- Prof. Wen and Prof. Chau supervised me and gave suggestions on my ideas, model, and experiments during the process of the research.
- I prepared the manuscript drafts. The manuscript was revised by Prof. Wen and Prof. Chau.

15/08/2023

. . . . . .

Date

Zhiyuan Yang

. . . . . .

Zhiyuan Yang

Acknowledgements

I'd like to express my profound gratitude to Prof. Wen and Prof. Chau for their meticulous guidance and thoughtful assistance. Their expert knowledge was instrumental in the successful completion of Thesis. I gained invaluable insights and experience from them.

Abstract

The use of DeepFake technologies to create hyper-realistic faces has sparked serious security concerns. Recent advances on DeepFake detection showed promise on algorithm generalization to unseen manipulation methods by identifying high-level semantic irregularities. However, the extracted features are not always robust, as the sample variations such as different motion magnitudes can easily degrade the feature-vector representations of their semantic information. In this work, we propose DTNet, a novel deep method that further regularizes feature learning toward more robust DeepFake Detection. To be specific, the proposed DTNet contains Deviation Regularization that penalizes samples with deviated motion magnitudes in the loss function, and Temporal Continuity Preservation, which helps keep and learn patterns of temporal continuity in feature space regardless of motion magnitudes. Experimental results show that our method effectively mitigates the impact of motion magnitudes on feature vectors, thereby improving the generalization ability.

Table of Contents

List of Figures

List of Tables

1. Introduction

In the ever-evolving landscape of technology, the introduction of Deepfake has sparked a global conversation surrounding the blurred lines between reality and illusion. These sophisticated manipulations of media content have created a world where seeing is no longer necessarily believing. At its core, Deepfake refers to the use of artificial intelligence (AI) to create hyper-realistic alterations of digital images, videos, and audio recordings, effectively enabling the synthesis of entirely new and often misleading content. This phenomenon has become increasingly prevalent with the rapid advancements in AI and machine learning, raising critical questions about the implications of such technology on our society, politics, and ethics.

1.1 Introduction to DeepFake

The term "Deepfake" is derived from the combination of two words: "deep," referencing deep learning, and "fake," denoting the creation of counterfeit or deceptive content. This concept was first introduced in the late 2010s, as researchers and technologists began to explore the capabilities of AI algorithms to generate synthetic media. They utilized neural networks, specifically generative adversarial networks (GANs) [1], to effectively "learn" from vast amounts of data and reproduce images, videos, and audio that closely mimic real-world counterparts. As the technology advanced, it became easier for even non-experts to create convincing Deepfakes, leading to a proliferation of this content across the internet.

One of the most well-known applications of Deepfake technology is in the creation of manipulated videos, wherein the face of a person in the original footage is replaced with the likeness of another individual. This has been popularized through celebrity face-swaps and impersonations, allowing viewers to witness surreal scenarios that were once considered implausible [2]. Moreover, Deepfake has also been utilized for voice cloning, enabling the replication of a person's speech patterns and tonal nuances, creating seemingly

authentic audio clips that can be used for various purposes [3].

While Deepfake technology has undeniably unlocked a world of creative potential in fields such as entertainment, advertising, and art, it also poses significant risks and challenges. Concerns over the malicious use of Deepfakes have grown substantially, with the potential to spread disinformation, manipulate public opinion, and even compromise the credibility of journalism. Fabricated videos and images have already been used in political smear campaigns, causing confusion and undermining trust in institutions [4]. Additionally, the rise of Deepfake pornography has brought forth a new dimension of privacy invasion and harassment, with victims often being unaware of the existence of such content until it has already circulated widely [5].

The increasing prevalence of Deepfake technology has prompted an urgent need for countermeasures. Researchers and companies have begun to develop detection tools that employ AI and machine learning to analyze content and identify potential Deepfake elements. These tools, however, face an ongoing battle against ever-improving Deepfake algorithms, resulting in a constant arms race between the two sides.

1.2 Research Objectives

The rapid advancement and proliferation of Deepfake technology have highlighted the urgent need for effective countermeasures to identify and combat these digital forgeries.

We aim to design an end-to-end DeepFake detection system that are robust to unseen DeepFake methods. This is a challenging task. Firstly, it requires the system to generalize its detection capabilities beyond the specific techniques and algorithms it has been trained on, effectively adapting to the ever-evolving Deepfake technologies. Secondly, the challenge of deepfake detection has attracted significant attention from researchers, industry professionals, and academics alike, transforming it into a highly competitive field. As more experts

dive into this area, an increasing number of state-of-the-art methods have emerged, the rapid pace of innovation and increasing number of state-of-the-art methods developed can make it difficult for researchers to surpass its predecessors in terms of detection accuracy, robustness, and generalization.

At the beginning of my research, I adopted a distinct and novel approach that focuses on capturing the patterns of real data instead of explicitly identifying the features of manipulated content. To the best of my knowledge, no one has tried to detect deepfake only using real data in the training stage. The underlying assumption of this strategy is that authentic data exhibits certain inherent characteristics that are difficult to replicate perfectly in deepfakes, regardless of the specific generation technique used. By training the model to recognize and understand the nuances of only real data, it can be better equipped to identify deviations from these patterns, which may indicate the presence of deepfake content. This approach does not rely on any assumptions about specific deepfake methods, making it potentially more resilient to a wide range of manipulation techniques, including those that have not yet been developed or encountered.

However, I failed to get competitive results only using real data to train. I did not find appropriate baseline and I did not manage to build a baseline from scratch by myself. Then I tried to use both real and fake data to train.

A recent state-of-the-art method in deepfake detection, known as Lipforensics [6], has demonstrated promising results by focusing on the semantic information of lip motions, which has proven to be robust to a wide range of deepfake generation techniques. This approach aligns with my ideas of concentrating on features that are difficult to manipulate accurately and consistently, rather than relying on assumptions about specific deepfake methods. Inspired by the success of Lipforensics, I decided to build my deepfake detection model based on this method, with the primary goal of extracting accurate semantic information from lip motions to identify and distinguish deepfake content.

With deeper exploration of high-level information, I found that the existing techniques for detecting DeepFake videos by analyzing the semantic information of lip movements are not sufficiently reliable. This is because differences in the magnitude of motion can easily weaken the feature vectors that encode semantic information, despite the fact that semantic information should be consistent regardless of the magnitude of motion. To illustrate, people can convey identical semantic information through their lip movements (such as when they say the same words) with varying degrees of motion magnitude, whether that motion magnitude is large or small.

Finally, our research objective is to a method for extracting feature vectors of semantic information that are invariant to motion magnitudes in order to improve the generalization of DeepFake detection. Our idea is generally illustrated in Fig. 1.

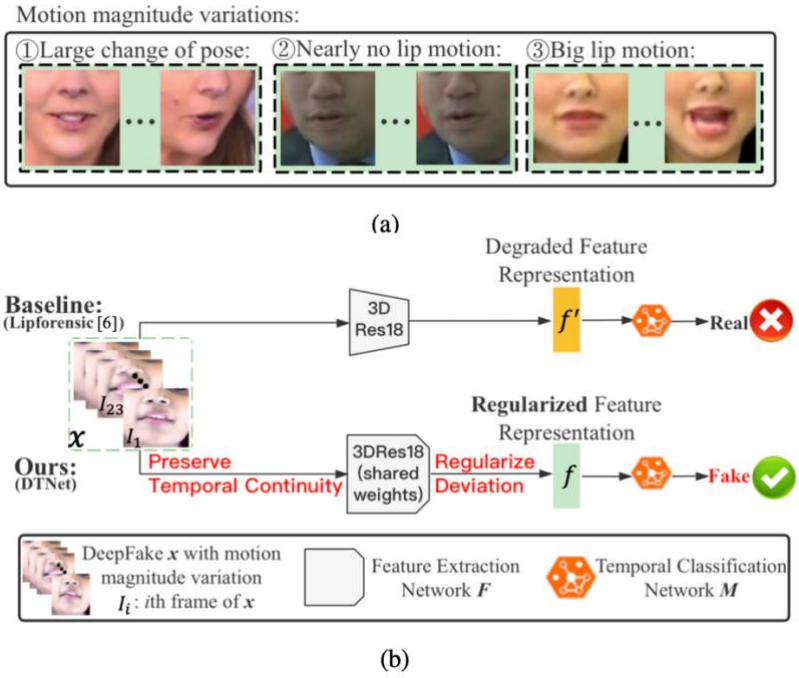

Fig. 1. Overall illustration of our method: (a) DeepFake samples of motion magnitude variations; (b) Comparison on dealing with DeepFakes of motion magnitude variations between Baseline Lipforensic (top) and ours (bottom)

Fig.1(a) shows three types of motion magnitude variations in DeepFakes, namely a large change of pose, nearly no lip motion and big lip motion. These phenomena led to big deviations in the feature space of semantic information

and thus impaired the DeepFake detection results in current methods. In Fig.1(b), we compare the baseline with our method when dealing with a DeepFake sample with motion variations like big lip motion shown in Fig.1(a). In the upper flow of Fig.1(b), the baseline [6] gets an input x and extracts lip motion representation f' through a feature extraction network F (3DResnet18). And then use another network M for classification. Unfortunately, the baseline tends to classify the DeepFake sample as real. However, our method shown in the lower part can successfully classify the DeepFake sample as fake. We propose Deviation regularization and Temporal continuity preservation Network **DTNet** to regularize the feature learning on semantic information to achieve it. Both are highlighted in red.

2. Literature Review

We review the generation and detection of DeepFakes in this section. Firstly, we generally describe four main types of DeepFake generation methods. Secondly, we introduce four primary DeepFake datasets and carefully investigate how fake videos are generated in each dataset. Next, we compare the generation and sophistication of each dataset in a table. Moving on, we explore current DeepFake detection methods and how researchers have exploited low-level and high-level anomalies caused by DeepFake generation for detection purposes. Finally, we compare the detection methods in terms of the information destroyed by generation, their catered generation method, and their limitations in a table.

2.1 Types of DeepFake Generation

Generally, there are four types of DeepFakes generation methods based on their objectives.

The first one is Face Replacement which swaps the face of the real image with another face [7]. Below Fig. 2 shows some examples of face replacement.

Fig. 2. Examples of Face Replacement [7]

The second one is Face Reenactment which transfers the expression of the source face to the target face while maintaining the identity of the target

person [8] [9]. Below Fig. 3. is an example of face reenactment.

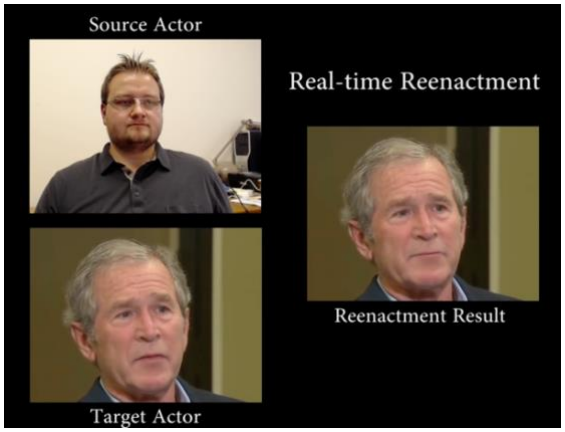

Fig. 3. An Example of Face Reenactment [8]

The third one is Facial Attribute Editing which alters the facial attributes of a person while preserving the identity of the person [10][11]. Below Fig. 4. is an example of facial attribute editing.

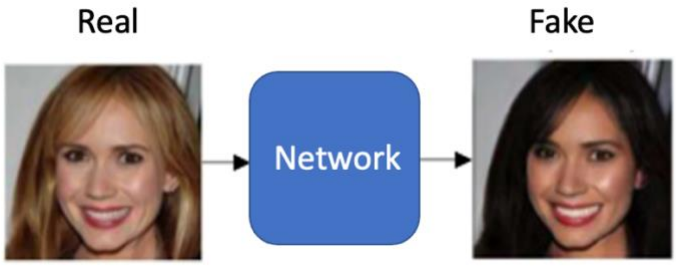

Fig. 4. An Example of Face Attribute Editing [11]

The fourth one is Face Synthesis which creates faces by mixing styles from different faces. Below Fig. 5. shows examples of face synthesis [12].

Fig. 5. Examples of Face Synthesis [13]

Face Replacement is more elementary compared to others and we can perceive the fake images easily with our eyes. However, some datasets applied post-processing techniques to smooth the boundary contour and eliminates the inconsistency and flickering of the swapped face [14]. Real-time face reenactment like Face2Face [9] is advanced and human can barely differentiate between real and fake. Although some are advanced, all current methods manipulated pristine videos frame by frame, which probably left inter-frame inconsistencies. And I try to exploit these inter-frame inconsistencies.

2.2 DeepFake Dataset

We train and evaluate our model in the following four datasets. It is important to understand how the fake videos are generated in the datasets. Generally, the majority of samples in current DeepFake datasets are generated by autoencoders. For our training dataset, FaceForensics++ [15] used Face2Face [9], NeuralTextures [16], FaceSwap [7], and DeepFakes [17]. For the testing dataset: CelebDF [14] used the autoencoder with post-processing techniques, FaceShifter [18] used the attribute editing encoder. DFDC [19] used eight different methods and most of the samples are generated through autoencoders. We elaborate these datasets first and then summarize the datasets in their Deepfake generation categories mentioned in Section 2.1.

2.2.1 FaceForensics++

FaceForensics++ [15] contains 4 different kinds of fake videos manipulated from the same set of real videos.

Firstly, shown in Fig. 3, Face2Face [9] was capable of transferring the facial expressions from a source video to a target video in real-time while preserving the other facial features and the background of the target person. The method involved analyzing and extracting information such as pose, illumination, expression, and identity of both the source and target actors on a frame-by-

frame basis. Initially, a 3D model was created using the first frames of each video to establish a temporary face identity, which was then used to track facial expressions in the subsequent frames. This approach was sophisticated and not reliant on deep learning methods.

Secondly, shown in Fig. 6, NeuralTextures [16] was also a reenactment method. They employed expression transfer to generate a modified UV map of the target actor, which matched the expression of the source actor. This UV map was used to sample from the target actor's neural texture. Additionally, they provided a background image to the neural renderer, resulting in the final reenactment output.

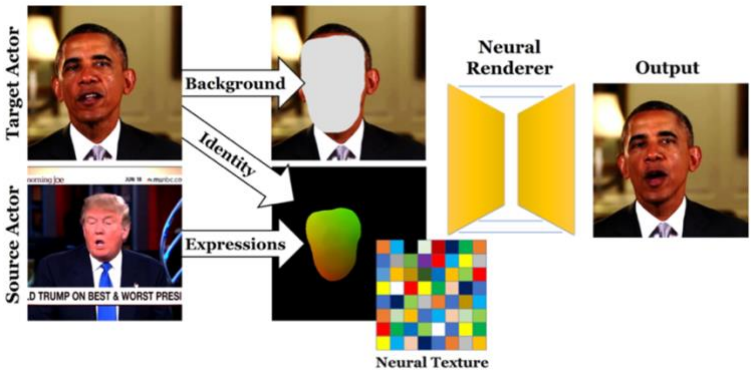

Fig. 6. Overview of NeuralTextures [16]

Thirdly, shown in Fig. 2, FaceSwap [7] involved first identifying a few key points on the face, from which the face region was extracted. These points were then used to fit a 3D template model that utilized blendshapes. The model was then projected onto the target image, and the difference between the projected shape and the localized landmarks was minimized by using the textures of the input image. The rendered model was then blended with the image, and color correction was applied to achieve the result.

Fourthly, shown in Fig. 7, DeepFake [17] was based on two autoencoders with a shared encoder that were trained to reconstruct training images of the source face A and the target face B, respectively. The architecture represented how the majority of fake images were generated in current datasets. To create a fake image, the trained encoder and decoder of the source face A were applied to

the target face B. The autoencoder output was then blended with remaining parts of the image through Poisson image editing.

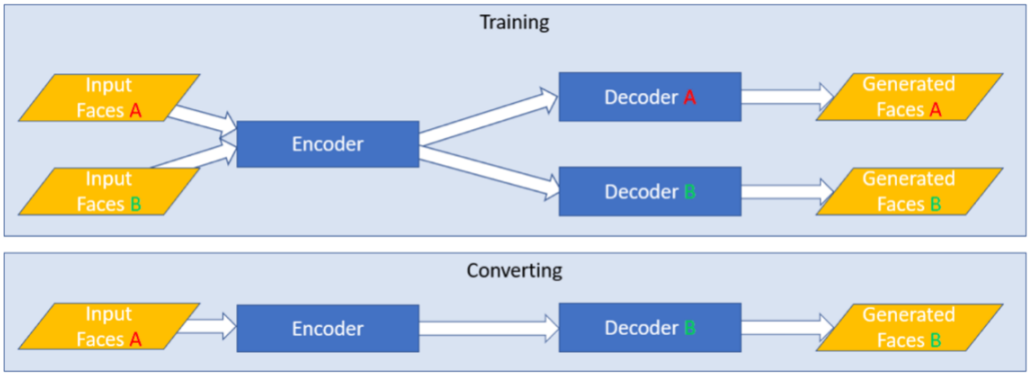

Fig. 7. Overview of DeepFake Autoencoders

2.2.2 CelebDF

Basically, CelebDF [14] generated fake images through Autoencoders similar to Fig. 7. and applied post-processing techniques like color correction, mask correction, and temporal correction. These post-processing techniques each of which is catered for a specific artifact left by DeepFake generation algorithms.

Below, Fig. 8. shows an example of color correction. In the training stage, they randomly perturbed colors of faces and also added color transfer from the real faces.

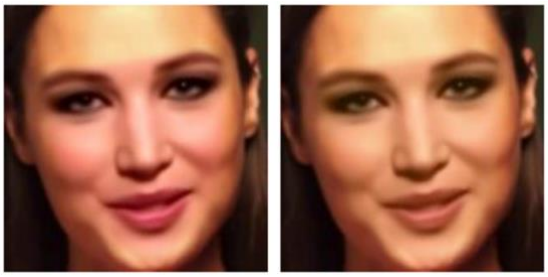

Fig. 8. Fake images without (left) and with (right) color correction.

For mask correction, they interpolated facial landmarks when generating fake images to make the boundary of mask invisible, shown in Fig. 9.

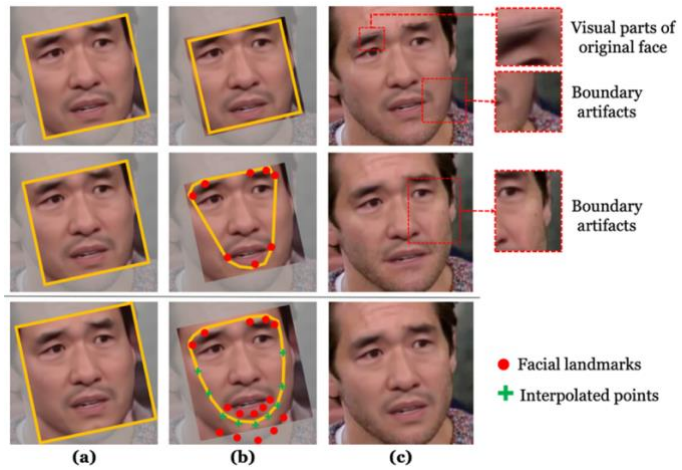

Fig. 9. Mask generations in other datasets (top two rows) and CelebDF (the last row). (a) the warped synthesized face overlaying the target's face. (b) mask generation. (c) final generated fake image [14]

For temporal correction, they filtered sequences of facial landmarks through a Kalman smoothing algorithm to reduce imprecise variations of landmark locations.

2.2.3 FaceShifter

We can regard FaceShifter [18] as an improved version of DeepFake autoencoders. It was composed of an identity encoder, a multi-level attributes encoder, and an AAD-generator. The AAD-generator integrated information of identity and attributes in multiple feature levels. It successfully addressed the defective lighting, face shape mismatch and blurred pixels, shown in Fig. 10.

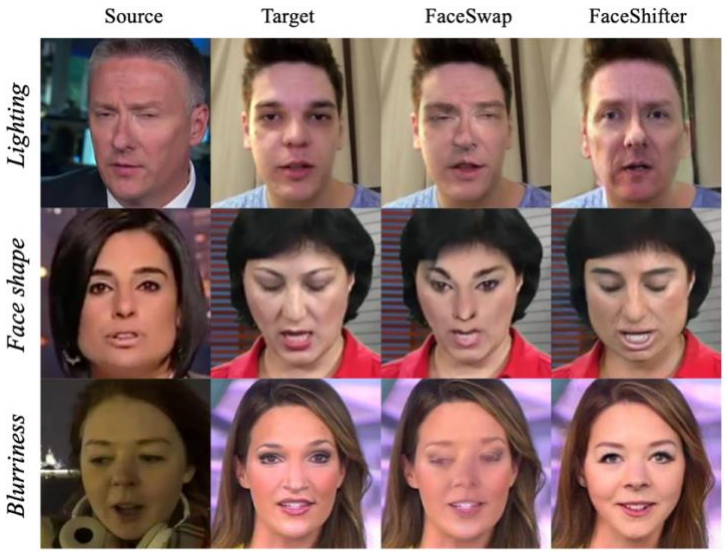

Fig. 10. Comparison of fake image generation between FaceSwap and FaceShifter. Three rows show issues of lighting, face shape and blurriness, which were addressed by FaceShifter (in the last two columns) [18]

2.2.4 DFDC

The majority of face-swapped videos in DFDC [19] were created with DeepFake Autoencoders, similar to DeepFake in FaceForensics++ and also applied multiple challenging data augmentations techniques. Unlike FaceForensics++, we did not know each fake video was generated by which generation method specifically.

Aside from DeepFake Autoencoders, DFDC contains samples generated from seven other algorithms and under multiple image background settings. For example, some used non-deep learning morphable mask model to transfer face action with personalized bilinear regression [20]. Some applied StyleGAN to project a fixed identity on the latent space of the face [21]. Some developed 4 GAN models to incorporate face swapping and reenactment through segmentations of face areas, inpainting and blending [22].

Different from other datasets, DFDC randomly applied multiple data augmentation methods to 70% of the generated videos, shown in Fig. 11.

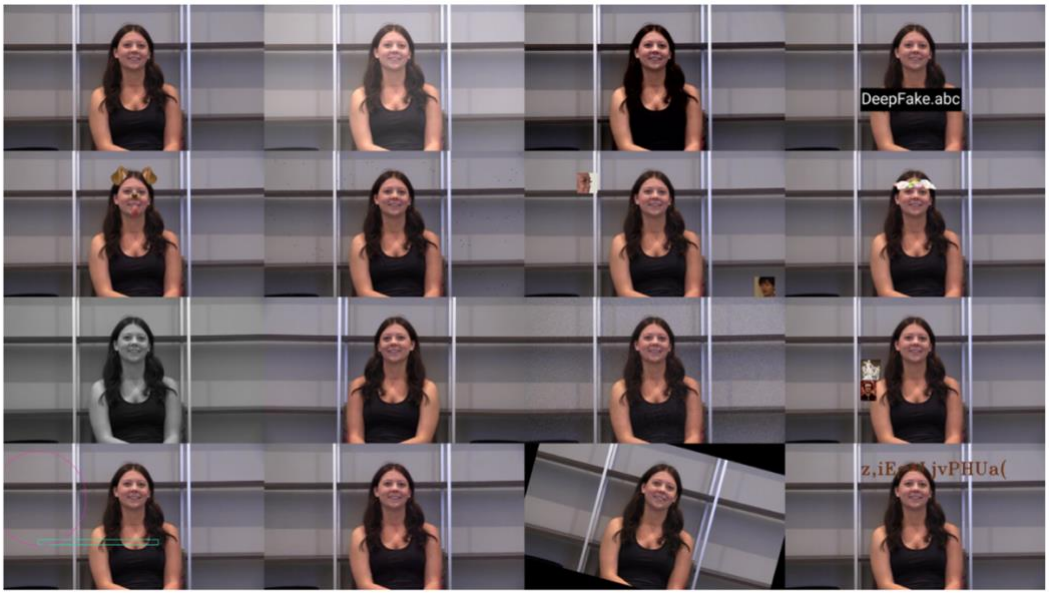

Fig. 11. A sample frame containing the various augmentations that were applied to the videos. Top row (from left to right): original, brightness, contrast, a logo overlay. Second row: dog filter, dots overlay, faces overlay, and flower crown filter. Third row: grayscale, horizontal flip, noise, and images overlay. Bottom row: shapes overlay, encoding quality level change, rotation, and text overlay. Not pictured: blur, framerate change, audio removal, and resolution change [19].

What's more, DFDC also contains more challenging samples like extreme lightning and extreme pose of the head, shown in Fig. 12.

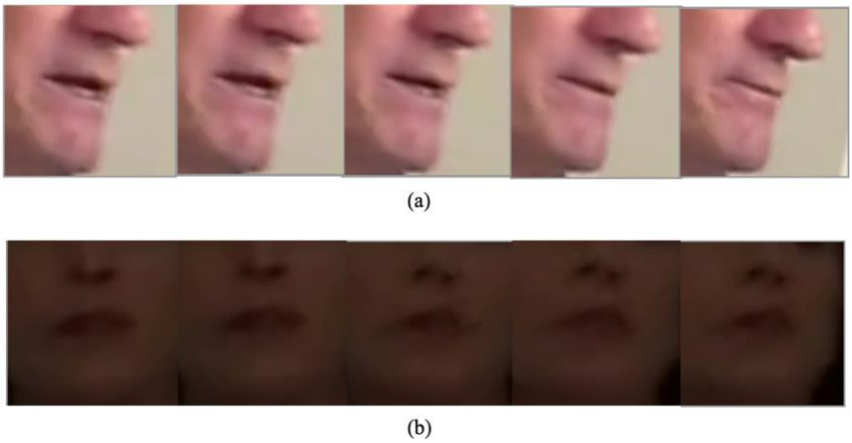

Fig. 12. Difficult samples in DFDC: (a) head with extreme pose, (b) extremely low lightning

Unlike most datasets contain samples with similar qualities or explicitly split samples of different qualities. DFDC contains sample of varying qualities. Like

Fig. 13 shown below, where each row corresponds to a method and each column shows increasing quality from left to right.

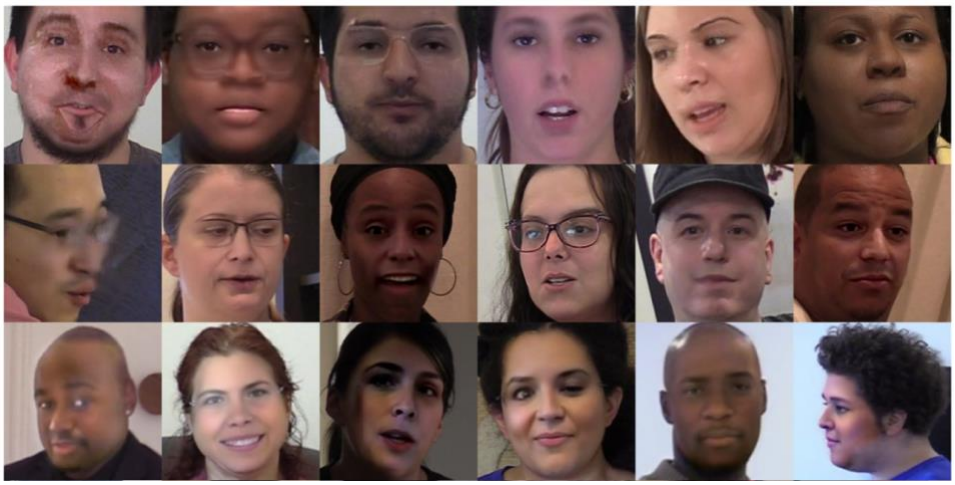

Fig. 13. Samples of varying qualities of different methods in DFDC

2.3 Summary of DeepFake Generation & Dataset

After carefully investigating four types of DeepFake generation and four primary DeepFake datasets. We listed the class split in Table. 1 and compared them in Table. 2. from the prospect of generation process, post-processing process and the level of sophistication.

Dataset	**#Videos**	
	Real	**Fake**
Face2Face (FF++) [9]	1k	1k
NeuralTextures (FF++) [16]	1k	1k
FaceSwap (FF++) [7]	1k	1k
DeepFake (FF++) [17]	1k	1k
CelebDF [14]	0.6k	5.6k
FaceShifter [18]	1k	1k
DFDC [19]	24k	105k

Table. 1. Summary of Class split in primary datasets

Dataset	Generation type	Model-based	GAN-based	Post-processing	Augmentation	Sophisticated
Face2Face (FF++) [9]	Face Reenactment	✓	✗	✗	✗	✓
NeuralTextures (FF++) [16]	Face Reenactment	✗	✓	✗	✗	✓
FaceSwap (FF++) [7]	Face Replacement	✓	✗	✗	✗	✗
DeepFake (FF++) [17]	Face Replacement	✓	✗	✗	✗	✗
CelebDF [14]	Face Reenactment	✗	✗	✓	✗	✓
FaceShifter [18]	Face Reenactment	✗	✓	✗	✗	✓
DFDC [19]	Face Replacement & Reenactment	✓	✓	✓	✓	✓

Table. 2. Summary of DeepFake generation in primary datasets

2.4 DeepFake Detection

A user study accuracy [15] indicated that Face2Face [9] and NeuralTextures [16] were particularly difficult to detect by human observers, as they do not introduce a strong visual change, shown in Fig. 14. As the technology for creating these videos becomes increasingly sophisticated, it becomes more challenging for humans to distinguish between real and fake videos. This has spurred the development of more advanced methods for detecting DeepFakes by examining them for telltale signs of manipulation. the battle between those who create DeepFakes and those who seek to detect them is currently in a state of stalemate.

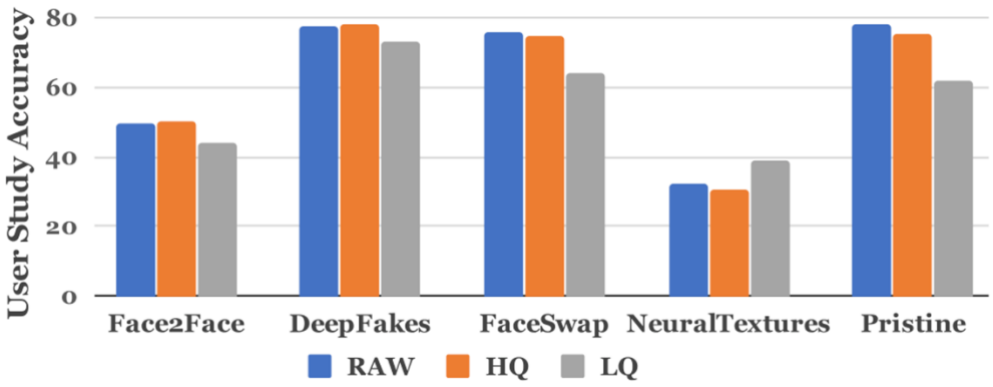

Fig. 14. Accuracy to differentiate DeepFake by human. (RAW: video of raw quality, HQ: video of high quality video, LQ: video of low quality) [15]

In this section, we are going to investigate current DeepFake Detection methods. Researchers have observed some anomalies caused by DeepFake generation and exploited them for DeepFake detection. We view these anomalies from two aspects: low-level and high-level. Low-level anomalies are visually perceivable anomalies in spatial space, frequency space and temporal space or other fused feature space. Also, DeepFakes may have high-level anomalies as DeepFake generation could destroy the semantic information in face motions, expressions, or identities.

2.4.1 Detection through Low-level Anomalies

Deepfake technology has raised concerns about the authenticity of images and videos, and detecting these forgeries has become increasingly challenging. One approach to Deepfake detection has been to look for low-level anomalies, which are visually perceptible anomalies in spatial space [23], frequency space [24-26] [31], temporal space [27-29], or other fused feature space [30] [32-35]. Previous models have focused on low-level anomalies, such as blending anomalies around boundaries [23], patterns in DCT spectra [24], or strange motion vectors through optical flow [27]. Some models have even fused information from color and frequency domains [30].

However, models focusing on low-level anomalies have not performed well in detecting unseen DeepFake generation methods. While low-level anomalies can be effective for detecting specific types of Deepfake artifacts, they may not capture the global properties of an image that can distinguish it as a forgery.

For spatial anomalies, elementary DeepFake generation methods extracted the source face and blend it into the target face, which left blending anomalies around the boundary. Some authors observed that most existing face manipulation methods shared this common step, i.e., blending the altered face into an existing background image. The Face X-ray [23] of an input face image is a grayscale image that reveals whether the input image can be decomposed into the blending of two images from different sources. It did so by showing the blending boundary for a fake image and the absence of blending for a real image. However, the assumption that the existence of a blending step is too strong for all DeepFake generation methods. Many advanced DeepFake generation methods did not rely on blending, in which scenarios Face X-ray may not get good results.

Aside from spatial anomalies, researchers also exploited anomalies in frequency space. Frank et al. [24] leveraged patterns in DCT spectrums. The authors observed that DeepFake images are often generated using generative adversarial networks (GANs), which introduced high-frequency artifacts that are distinct from those found in real images. Building on this insight, they proposed

CNN to analyze the frequency spectrum of an input image and distinguish between real and fake images based on the presence or absence of high-frequency artifacts. However, it may not be generalizable to other types of image manipulations, as the model is specifically trained to detect high-frequency artifacts introduced by GANs. Similar papers like [25] proposed to extract and detect DeepFake through DCT coefficients. From another aspect, Y. Luo et al. [26] tired to detect DeepFake by high-frequency noise. They assumed that noises were high-frequency signals that capture random fluctuations in brightness or color data. The distribution of image noise was affected by the digital camera's image sensor and internal circuitry. DeepFake generation would destroy the pattern of high frequency noise. The proposed method involved applying a high-pass filter to the down sampling of multiple RGB feature maps to extract high-frequency features and then using a deep neural network to analyze these features to distinguish between real and fake images. However, the essential high-pass filters were carefully handcrafted by human, which may be tuned to overfit only several DeepFake generation methods but could not generalize to unseen methods. Recent method [31] generated adversarial examples to improve the generalization ability of deepfake detection models. However, it assumed that blending existed in the Deepfake generation process, which restricted its generalization to certain types of generation methods.

For temporal space, Amerini et al. [27] found strange motion vectors through optical flow of fake frame sequences. The method was rather straight. They computed optical flow of consecutive frames and input them to an CNN for classification. Although the detection accuracy was not competitive, this is the first time that optical flow-based CNN method was applied in DeepFake detection. It also indicated that simple feature may not be sufficient to achieve good performance in DeepFake detection. Furthermore, methods that tried to exploit inter-frame conflicts [28] showed promising. The proposed approach utilized CNN to extract spatial features from video frames and RNN to capture temporal dependencies between the frames. Also, the authors found that face alignment could help improve the results and that a sequence of images was better than single frame input, both of which were widely used in subsequent

methods. To the best of our knowledge, all current deepfake generation methods generated fake images frame by frame, which made it highly probable that temporal inconsistencies existed across the frame. Recent method [29] achieved state-of-the-art performance by temporal incoherence that happened between the neighborhood frames. The authors argued that temporal coherence, which referred to the consistency of motion and appearance across frames, was an essential factor in detecting face forgery in videos. The proposed approach incorporated a temporal coherence module into the existing deepfake detection framework, which utilized both spatial and temporal information. The temporal coherence module was designed to capture long-term temporal dependencies in video frames and enforce consistency across frames. However, the application of transformer was more computationally expensive.

Additionally, some researchers exploited anomalies in fused feature space. The anomalies may not be as perceivable by human as previous ones. Two-branch [30] fused information from color and frequency space. The spatial branch processed individual frames using a convolutional neural network, while the temporal branch used a recurrent neural network to analyze the temporal consistency between frames. The two branches were then combined to make a final deepfake prediction. But it may be overfit to seen deepfake generation methods and it needed further evaluation on other datasets. C. Shen et. al [32] observed that existing DeepFake detection methods often relied on global features or pixel-level information, which may not capture the complex relationships between different regions of the face. To address this limitation, the authors proposed a new method that leverages local relation learning to capture the relationships between local regions of the face through RGB and DCT fusion. However, the classification was based on a specific kind similarity map of a specific DeepFake generation method. When a novel DeepFake generation method appeared, it easily failed to output a new type of similarity map, which leads to impaired classification results.

Recent method [33] also tried to detect deepfake from correlations between local regions. The authors assumed that there were self-consistencies between

local regions of real image, which could easily be destroyed by DeepFake generation methods. The proposed method involved training a generative network to generate adversarial examples by modifying real images or videos. The adversarial examples were then used to train a detection model to distinguish between real and fake images or videos. Although it got competitive results, it may be because that the majority of DeepFake samples in testing datasets were used similar DeepFake generation methods as the training adversarial samples. As the training and testing results used similar DeepFake generation methods, the results may not support that the model had good generalization ability to novel generation methods. Lastly, multi-modal data like audio visual consistency were used for DeepFake Detection too [34] [35]. They observed that existing DeepFake detection methods mainly focused on visual features and neglected the role of audio in DeepFake generation. Basically, this kind of methods applied mel-scale spectrogram to represent audio information and extracted words through lip motions. The model involved training separate visual and auditory models and then combined the output of both models to form a synchronization detection. The auditory model also relied on high-level anomalies, which we will elaborate on Section 2.4.2. However, it may be computationally expensive due to the use of both visual and auditory models, and it lacked the validation of the accuracy of auditory models.

Some general effective techniques in the computer vision field were also applied to deepfake. H. Zhao et.al [36] applied a multi-attention module to detect DeepFake. The authors drew inspiration from the fine-grained classification problem, where species with small and local differences are differentiated from each other. They proposed to model deepfake detection as a special fine-grained classification problem with two categories and present a multi-attention network for deepfake detection. The proposed network consisted of multiple spatial attention heads that predicted multiple attention maps and enhanced the textural features obtained from shallow layers. The authors aggregated low-level texture features and high-level semantic features as the representation for each local part and then pool the feature representations of each local part independently by a bilinear attention pooling layer, which was fused as the

representation for the whole image. However, the model was not easy to train. What's more, the transformer has also been applied in DeepFake Detection [37]. The authors proposed a novel approach that leveraged transformer models to capture the relationships among pixels and effectively explore regions of different scales in images. The proposed Multi-modal Multi-scale Transformer (M2TR) followed a two-stream architecture that operated on patches of different sizes to detect local inconsistencies in images at different spatial levels. The frequency stream adopted learnable frequency filters to filter out forgery features in the frequency domain, which complement RGB information. The authors further introduced a cross-modality fusion block to combine the information from both streams effectively. However, using transformer did not boost the performance very much comparing to its more expensive computational cost. More simple and efficient transformer-based detection method is needed.

2.4.2 Detection through High-level Anomalies

DeepFakes can be created with through various generation methods, making it increasingly challenging to detect them using one robust model. One approach to DeepFake Detection has been to look for high-level anomalies, which may be present in Deepfake images due to the destruction of semantic information in facial motions, expressions, or identities during the generation process. Models tried to detect high-level anomalies in fake video has been proved a great help to boost model generalization to novel methods. However, research papers focusing on high-level anomalies are much less than those focusing on low-level anomalies. Future work in Deepfake Detection should continue to explore high-level features. We will review some papers focusing on high-level anomalies below.

One promising approach is to focus on detecting high-level anomalies in identity, that are difficult to manipulate and are unique to each individual [38] [39] [40]. Previous method [40] leveraging simple identity of the person through the 4 clips of the same video did not boost generalization ability. Recently, D.

Cozzolino et al. [39] proposed a new approach called ID-Reveal, which learned temporal facial features specific to how a person moved while talking. It was more complex definition of identities than the previous method that simply defined identity as who the person is. The approach included three primary elements: a tool to identify facial characteristics, a system to recognize biometric irregularities over time, and a network that attempted to determine individualized movements from the expressions of another person. The metric learning coupled with an adversarial training strategy and high-level semantic features made it resilient to post-processing techniques. However, the authors mainly compared their results with other methods published before 2019, they should include more recent methods. Similar as [40], X. Dong [38] used simple identity definition but equipped with Identity Consistency Transformer (ICT), incorporating a consistency loss for identity consistency determination, making it more robust to various types of image degradation forms, including deepfake videos. The proposed ICT method trained a model to learn a pair of identity vectors, one for the inner face and the other for the outer face, by designing a Transformer such that the inner and the outer identities could be learned simultaneously in a seamlessly unified model. The paper also showed that the proposed ICT could be easily enhanced with additional identity information when such information was available, as was the case with celebrities. However, only a few datasets labeled their samples with identity of the person. Also, it needs to include more recent results for comparison.

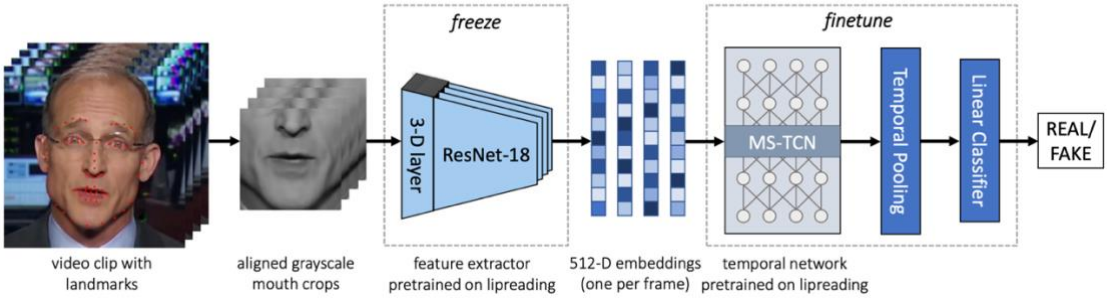

Fig. 15. Overview of Lipforensics [6]. After preprocessing, 25 frames of cropped mouth are input into the network. These crops undergo a frozen feature extraction process through a ResNet-18 that includes an initial 3-D convolutional layer. The feature extractor was pre-trained on lipreading dataset, leading to

embeddings attuned to mouth movements. Following this, a multi-scale temporal convolutional network (MS-TCN), which was also trained on lipreading, gets fine-tuned to detect fake videos by high-level semantic irregularities in mouth movement.

Another approach is to leverage high-level anomalies in semantic meaning of the lip motions. Lip motions, also known as lip movements or lip gestures, can convey a range of semantic meanings depending on the context in which they occur. The primary function of lip movements is to produce speech sounds. The position and movements of the lips help to shape the sounds that we produce when we speak, and can indicate the pronunciation of vowels and consonants that make up words, as well as the emphasis, intonation, and emotion behind what we're saying. Recent method Lipforensics [6] assumed DeepFake generation could destroy such semantic meaning of lip motions. They achieved the current state-of-the-art performance and demonstrated good generalization ability to unseen DeepFake generation methods. Shown in Fig. 15, the approach consisted of first pre-training a spatial-temporal network to perform visual speech recognition (lipreading), thus learning rich internal representations related to natural lip motion. A temporal network was then fine-tuned on fixed mouth embeddings of real and forged data to detect fake videos based on mouth movements without overfitting to low-level, manipulation-specific artifacts.

However, after implementing Lipforensics [6] on DeepFake samples across different datasets. We found that the method based on extracting the semantic information of lip motions were not robust enough. The AUC of the model dropped 3% when we only tested it with samples of big motion magnitudes and dropped 5% with samples of small motion magnitudes. Although Lipforensics could get the feature space representing semantic information to some extent by pretraining the model first with Lipreading dataset [41], the results indicated that variations of motion magnitudes could easily degrade the feature-vector representations of the semantic meaning, but the semantic information should be invariant to motion magnitudes. For example, the same semantic information could be conveyed (e.g., people say the same words) with either a big motion

or a small motion.

Hence, we try to mitigate the negative influence posed by motion magnitudes. We propose Deviation Regularization and Temporal Continuity Preservation to regularize the learning of semantic information, and thus make the feature-vector representations more robust.

2.5 Summary of DeepFake Detection

With DeepFakes being created using various generation methods, it has become increasingly difficult to detect them using a single robust model. We have carefully investigated several deepfake detection methods based on either low-level anomalies or high-level anomalies. In this section, we also compared them in the aspects of information destroyed by generation (assumed by the authors), whether they were cater to a specific kind of generation method and their limitations in Table 3.

Reference	Information destroyed by generation	Cater to specific generation methods	Limitations
[23][31]	Blending boundaries	✓	Many advanced DeepFake generation methods did not rely on blending.
[24] [25]	DCT coefficients	✓	Not all fake samples have high-frequency anomalies introduced by GANs.
[26]	Camera fingerprints	×	Handcrafted parameters for essential high-pass filter.
[27] [28] [29] [30]	Inconsistencies across frames	×	[27] and [28]: not good AUC. [29]: computationally expensive. [30]: may overfit and needs evaluations on other datasets.
[32] [33]	Correlations between local regions	✓	Specifically fit to a kind of generation methods and thus lacks generalization.
[34] [35]	Audio-visual consistency	×	Needs to validate the accuracy of auditory models.
[36] [37]	Patterns in color and frequency space	×	Not easy to train and computationally expensive.
[38] [39] [40]	Identity	×	Should include more recent methods for comparison. Only A few available datasets are labeled with identity.
[41]	Semantic meaning of lip motions	×	Not robust to motion magnitudes.

Table. 3. Summary of DeepFake detection methods

3. Methodology

We aim to build a DeeepFake detection model that is both robust and generalizable by differentiating between natural and abnormal mouth movements, meanwhile, the feature vectors of semantic information are robust to motion magnitudes. The assumption is that irregularities in lip motions exist in fake videos regardless of the DeepFake generation method.

Firstly, same as Lipforensics [6], we pretrain train a convolutional neural network (CNN), which includes a spatio-temporal feature extraction component and a subsequent temporal convolutional network, for lipreading tasks. We anticipate that this training will generate internal representations that can detect irregular mouth movements within a high-level semantic context, as low-level patterns may not be adequate for accomplishing the task.

Then based on the Lipforensics, we proposed Deviation Regularization and Temporal Continuity Preservation to regularize the feature learning and thus mitigate the negative effect posed by motion magnitudes. In Sec. 3.1, we explain the method of using a video-level inter-frame distance in feature space to represent motion magnitudes. In Sec. 3.2, we propose deviation regularization to penalize big variations in motion magnitude. Lastly, in Sec. 3.3, we introduce the idea of preserving temporal continuity in feature space by ensuring that feature vectors of consecutive frames do not differ too much. These three techniques are essential to the proposed deepfake detection model and are explained in detail in the following sections.

3.1 Inter-frame Distance in Feature Space

In Sec.3.1, we first find an index that can represent motion magnitudes. The idea is derived heuristically from that inter-frame difference in the image space can represent motion magnitudes and we transfer it into the feature space. So we define video-level inter-frame distance in feature space, i.e. $D_v(x)$.

As indicated in the previous sections, we need to deal with variations of motion magnitudes. And the inter-frame difference that refers to the pixel-wise difference between two frames in color space can help represent these variations (e.g., big inter-frame differences in color space can represent big motion magnitudes). Hence, we regularize inter-frame distance in the feature space to reduce the influence posed by these variations.

In this section, we define the video-level inter-frame distance in feature space. Then in the following two sections, we will elaborate on how to regularize inter-frame distance in feature space to reduce the influence of different motion magnitudes on semantic information.

In feature space, we use the cosine similarity to measure the inter-frame distance in feature space and adjust the range to $[0,1]$ similarly as [32]. So the frame-level inter-frame distance $D(i,j)$ between $i-th$ frame and $j-th$ frame is:

$$D(i,j) = 1 - \frac{cos < f_i, f_j > +1}{2} \qquad (1)$$
$$f_i = F(I_i)$$

where F is the feature extraction network, I_i is the $i-th$ frame of the input video x, and f_i is the feature vector of the frame I_i.

Then the video-level inter-frame distance $D_v(x)$ is computed by averaging $D(i,j)$ of all frames within a video:

$$D_v(x) = \frac{1}{k^2} \sum_{i=1}^{k} \sum_{j=1}^{k} D(i,j) \qquad (2)$$

where x is the input video of k frames.

3.2 Deviation Regularization

In Sec.3.2, we propose deviation regularization to regularize samples with motion magnitudes. We define $\sigma(X)$ from $D_v(x)$. When samples with motion magnitude variations appear, $\sigma(X)$ becomes big. Then we penalize big $\sigma(X)$ in

our loss function to regularize motion magnitude variations.

A big change of a head poses or nearly no lip motion could lead to large deviations of inter-frame differences. Deviation regularization penalizes samples of these large deviations when they are beyond a certain range. The deviation of inter-frame distance across population X is defined as $\sigma(X)$:

$$\sigma(X) = \sqrt{\frac{\sum_{x_i \in X}(D_v(x_i) - \mu)^2}{N-1}} \qquad (3)$$
$$\mu = \frac{\sum_{x_i \in X} D_v(x_i)}{N}, X = \{x_1, x_2, .., x_N\}$$

where N is the number of videos and x_i is the $i-th$ video.

We set a boundary γ to $\sigma(X)$, and the deviation regularization only contributes to the whole loss function when $\sigma(X)$ is larger than the boundary. Hence, the loss function L is defined as:

$$L = L_{ce} + \lambda \max(\sigma(X) - \gamma, 0)$$
$$L_{ce} = -\sum [y_i \log \hat{y_i} + (1 - y_i)\log(1 - \hat{y_i})] \qquad (4)$$
$$\hat{y_i} = M(f_1, f_2, ..., f_k)$$

where L_{ce} is the cross-entropy loss, λ is the weight of regularization, and γ is the boundary coefficient. We passed the extracted feature vectors through another temporal classification network M to obtain the final predicted probability $\hat{y}$ The training process minimizes *L* over all parameters in the feature extraction network *F* and the temporal classification network *M*.

3.3 Temporal Continuity Preservation

In Sec.3.3, we propose to preserve temporal continuity in feature space. We also derive this idea heuristically from the image space. Temporal continuity can be described as the gradual change between consecutive frames. We transfer it into the feature space: temporal continuity means feature vectors of consecutive frames do not differ so much. And we obtain similar feature vectors for consecutive groups of frames by making the input of feature vectors do not differ so much. Because we use a shadow network with a limited discrimination

ability to extract feature vectors. When the input is similar, the output feature vectors should be similar too.

Temporal continuity in color space refers to the rule that moving areas change gradually and static areas remain the same between adjacent frames. Similarly in feature space, the temporal continuity may be preserved if there is no large distance between feature vectors of adjacent frames. However, in current methods, big motion magnitude made it hard for the feature space to preserve such temporal continuity, because it triggered a bigger $D_v(x)$ (defined in Eq.(2)), which meant that feature vectors of each frame differed so much.

Temporal continuity is important to detect DeepFakes as DeepFake generation could destroy this rule whereas real video follows this rule. To preserve such temporal continuity in feature space and thus force network *M* to learn the rule of temporal continuity in feature space, we propose to ensure the small distance between adjacent feature vectors f_i and f_{i+1} of adjacent frames I_i and I_{i+1} no matter the motion magnitude is big or small. As defined in Eq.(1) and Eq.(2), ensuring small distances is equivalent to ensuring the high cosine similarity between adjacent feature vectors.

We split the input into groups of frames and make adjacent groups have some same frames. And then we feed groups of frames parallelly into feature extraction networks of shared weights, shown in Fig. 16. Hence, we can rewrite the feature vectors with temporal continuity in Eq.(1):

$$F(I_{2i-1}, I_{2i}, I_{2i+1}, I_{2i+2}, I_{2i+3}) = f_i, \quad i \in \{1, 2, 3, .., 11\} \tag{5}$$

where *F* is the feature extraction network, f_i is the feature vector of ith group of frames, I_{2i-1} is the $(2i-1)th$ frame of the video clip *x*.

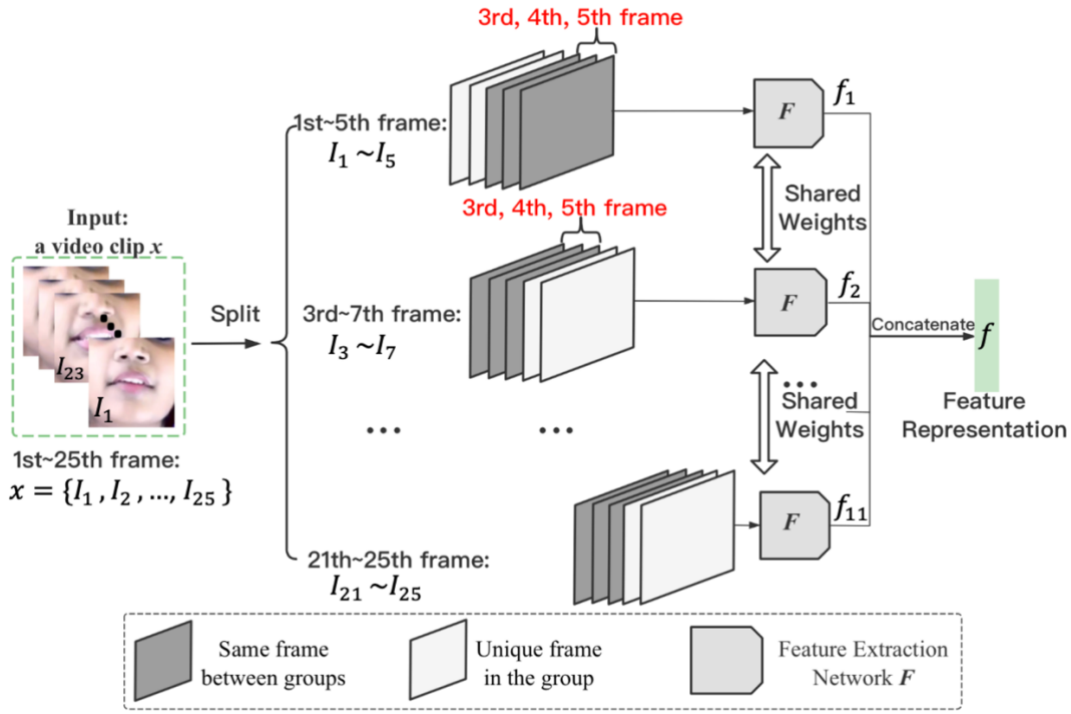

Fig. 16. The process of getting temporally continuous feature vectors

The method is on the basis that if two inputs to the network *F* are similar, which means input two frames have a low pixel-wise texture difference, then their outputs will also be similar, as *F* is a shallow convolutional neural network for feature extraction that has a limited discrimination ability. We also validate this prior basis in our model in Section 4.3.

After making adjacent groups have some same frames, parallel inputs to *F* contain a high percentage of the same frames. In other words, the same frames make sure that the inputs of *F* for f_i and f_{i+1} are similar enough, hence we can guarantee their outputs f_i and f_{i+1} have high enough cosine similarity.

As illustrated in Fig. 16, we split an input video of 25 frames into 11 groups of frames. Each group has five frames, and adjacent groups have three same frames. 11 groups of frames are fed into the feature extraction network *F* with shared weights in parallel. We get a feature vector for each group and concatenate them together for subsequent temporal classification through another network *M* defined in Eq.(4).

4. Experiments

4.1 Experiment Settings

Dataset: We train and test our model on the most commonly used benchmark dataset HQ (c23) version of FaceForensics++ (FF++) [15]. It contains 1000 real videos, and each of them is manipulated by four different DeepFake methods: Face2Face, FaceSwap, Deepfakes, and NeuralTextures. Furthermore, to evaluate the model generalization ability, we test our model on FaceShifter (FSh) [18] with 280 testing videos and CelebDFv2 (CDF) [14] with 518 testing videos.

Architecture: For network architecture, we use 3D-ResNet18 as feature extraction network *F*, and use Multi-scale Temporal Convolutional Networks [42] as network *M* to further extract and classify the feature vectors same as Lipforensics [6].

Preprocessing: First, the faces are detected and adjusted to match the average face. Then, a grayscale 96 × 96 area around the mouth is cropped and transformed. Our videos have 25 frames, so after random cropping, our input tensor is 25 × 88 × 88 × 1 in size. It's important to note that this size is comparable to the standard RGB frame input for many forgery detectors (which is typically 1 × 256 × 256 × 3) according to existing research.

Training Scheme: For the training scheme, the model is firstly pretrained in LRW [41], a dataset containing over 500,000 utterances spanning hundreds of speakers in various poses, and then is fine-tuned in DeepFake datasets, following Lipforensics [6]. Our models are trained with SGD. The learning rate is 10^{-4} and weight decay is 10^{-7}. In Eq.(4), the weight of deviation regularization λ is 0.1, and the boundary γ is 0.13. We terminate training when there is negligible improvement to the validation loss for 10 epochs.

Evaluation Metric: All results are expressed with AUC, which is commonly

used in Deepfake detection. It is robust to class imbalance, as it considers the true positive rate (sensitivity) and false positive rate across thresholds, providing a balanced view of model performance. In tasks like deepfake detection, where the tolerance for error rate is low, the consequences of false positives and false negatives are significant. Through AUC, both types of errors are reduced as much as possible, leading to a more reliable and trustworthy model.

4.2 Comparison with Previous Methods

We evaluate our model in the cross-magnitude setting, cross-manipulation setting and cross-dataset setting. Cross-magnitude setting means that we split the dataset according to its motion magnitude and conduct experiments on each part. Cross-dataset setting means that the training samples and testing samples come from two different datasets. Cross-manipulation setting means that the DeepFake generation methods in training and testing datasets are different, but the fake samples are manipulated from the same real samples. We adopt video-level AUC (area under Receiver Operating Characteristic Curve) as the evaluation metrics [6].

Cross-magnitude Testing Comparison: We train in FaceForensics++ (FF++) and test in FaceShifter (FSh). Table. 4. shows that we achieve a performance gain of 5% in samples with big motions, and a performance gain of 8%. Our methods successfully reduce the effect posed by motion magnitudes.

Methods	Motion Magnitude		
	Big	Small	Regular
Lipforensics[6]	0.88	0.86	0.91
Ours	**0.93**	**0.88**	**0.92**

Table. 4. AUC with different Motion Magnitudes

Cross-dataset Testing Comparison: We train our model in FaceForensics++ (FF++) and test it in another dataset CelebDFv2 (CDF). We compare our results with six recent methods. The second column of Table. 5. shows that our method has better generalization ability. It surpasses Lipforensics by 2.1% AUC, from

82.4% to 84.5%.

Methods	Cross-dataset AUC (CDF) ↑	Cross-manipulation AUC (FSh) ↑
Two-Branch [30]	0.767	-
Face X-ray [23]	0.795	0.928
Multi-task [43]	0.757	0.660
SLAE [31]	0.837	-
Multi-attention [36]	0.721	-
Lipforensics [6]	0.824	0.971
DTNet (ours)	**0.845**	**0.982**

Table. 5. Cross-dataset and Cross-manipulation testing results on CelebDF (CDF) [14] and FaceShifter (FSh) [18] when training on FaceForensics++ (FF++) [15] (We test SLAE [31] and Multi-attention [36] to get video-level AUC by ourselves, others are from [6]).

Cross-manipulation Testing Comparison: We train in FaceForensics++ (FF++) and test in FaceShifter (FSh). The third column of Table. 5. shows that our methods can generalize well to novel manipulation methods and surpass Lipforensics by 1.1% AUC, from 97.1% to 98.2%.

4.3 Ablation Study

We study the importance of Deviation regularization and Temporal continuity preservation in this section. We also elaborate on hyperparameter setting. Different from the experimental setting in Section 4.2, we pretrain all models with the first 50 classes (10%) of LRW for affordable computational cost. Other settings are the same.

Hyperparameter Setting: We have three manually set hyperparameters, namely γ, λ and temporal window size. The ablation study is shown in Table. 6.

Boundary coefficient γ: To determine γ in Eq.(4), we firstly compute the mean of $\sigma(X)$ of all samples in pretraining stage. The mean can be regarded as the typically normal value of deviation. When samples have much larger $\sigma(X)$ than

the mean, we need to penalize them. And we experimentally search around the mean to find the final value of γ.

Weight of deviation λ: λ is set empirically as a regularization to the loss function. We fine-tune it to make a balance with the cross-entropy term.

λ	0.01	0.05	0.1	0.15	1
AUC↑	0.70	0.70	0.72	0.71	0.61
γ	0.01	0.11	0.13	0.15	1
AUC↑	0.65	0.71	0.72	0.72	0.70

Table. 6. AUC with different λ and γ. $\lambda = 0.1$ when γ changes and $\gamma = 0.13$ when λ changes

Temporal window size of frame groups: We need to ensure that we have a fairly good lip motion representation f_i (defined in Eq.(5)). Otherwise, the subsequent classification based on f_i would be impaired. Too large window size of frame groups in Sec.3.3 is computationally expensive whereas too small may lead to inaccurate lip motion representations. As indicated in Lipforensics [6], five frames are sufficient to capture lip motions and achieve AUC around 0.89 in FaceShifter. To balance the computational cost and accuracy, we set the window size of each group in Sec 3.3 to five.

Methods	Cross-manipulation	Cross-dataset
	AUC (FSh) ↑	AUC (CDF) ↑
Lipforensics(Baseline) [6]	0.902	0.639
TCP	0.915	0.699
DR+TCP(Ours)	**0.923**	**0.721**

$\sigma(X)\downarrow$	$D_v(x)\downarrow$	
	Big Motion	Small Motion
0.015	0.1821	0.0840
0.014	0.0464	0.0322
0.011	**0.0432**	**0.0310**

Table. 7. Effect of Temporal Continuity Preservation (TCP) and Deviation Regularization (DR)

Effect of Better Generalization: The results in the second and the third column of the Table. 7. show that both Deviation regularization and Temporal continuity preservation are performant in improving the cross-manipulation generalization and cross-dataset generalization ability of the model.

Effect of Deviation Regularization: As we proposed in Section 3.2, Deviation Regularization aims to maintain $\sigma(X)$ in Eq.(3) within an acceptable range. We can see from the fourth column in Table. 7. that our methods can successfully decrease $\sigma(X)$, which indicates Deviation regularization is performant.

Effect of Temporal Continuity Preservation: In Section 3.3, we design Temporal continuity preservation to keep temporal continuity in feature space. Here, we validate it through $D_v(x)$ defined in Eq.(2). We cut two video clips from the same video: one clip has a big motion, and the other has a small motion. From the last two columns of Table. 7, we can see that the our method successfully guarantees small $D_v(x)$ when a big motion appears and thus keep the temporal continuity in feature space. We also test the number of the same frames from 0 to 4, where adjacent groups that have three same frames get the best result.

Visualizations of Feature-vector Representations:

We visualize the effect of the proposed methods by comparing the similarity map of feature vectors. For better visualization, the pixel $p(i,j)$ of ith row and jth column in the similarity map is defined from Eq.(1):

$$p(i,j) = \frac{cos < f_i, f_j > +1}{2} \qquad (6)$$

where f_i and f_j are the feature vectors defined in Eq.(5).

Fig. 17(a). and Fig 17(b). are two clips with a big lip motion and a small lip motion, respectively. Fig. 17(c). and Fig. 17(d). are similarity maps of these two

clips when testing in the baseline. We can see that the clip with a small motion gets higher similarity than that with a big motion, as the input frames are more similar (have low inter-frame pixel-wise difference), which validates our prior basis in Section 3.3. And Fig. 17(e). and Fig. 17(f). are similarity maps of these two clips when testing in DTNet (ours). We can see that similarity remains high, i.e. small $D_v(x)$, regardless of the motion magnitude, which justifies that feature vectors preserve the temporal continuity.

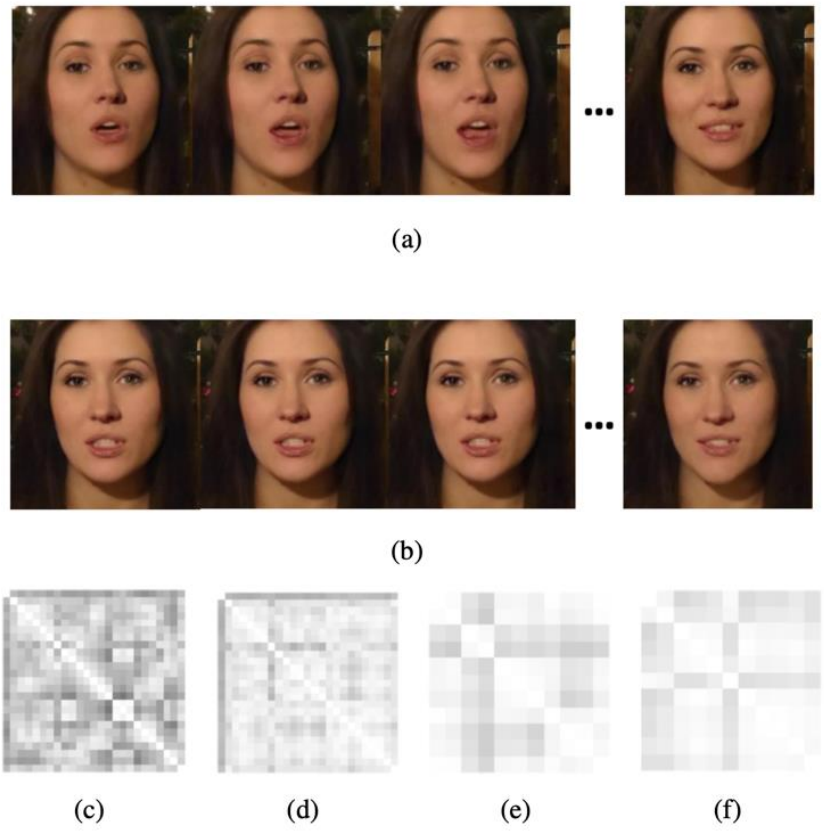

Fig. 17. Similarity maps of big motion and small motion clips from the same video: (a) a video clip of big motion; (b) a video clip of small motion; (c) similarity map of the big motion clip (baseline), $D_v(x)$: 0.1804; (d) similarity map of the small motion clip (baseline), $D_v(x)$: 0.0839; (e) similarity map of the big motion clip(ours), $D_v(x)$: 0.0578; (f) similarity map of the small motion clip(ours), $D_v(x)$: 0.0464.

We also visualize some samples which are wrongly classified by the baseline but correctly classified by DTNet (ours). Fig. 18(a), Fig. 18(d) and Fig. 18(g). are

three types of wrongly classified samples by baseline, but our model correctly classifies them. As mentioned in Fig. 1, the baseline did not perform well for some samples with big lip motion like Fig. 18(d), some with big changes of head pose like Fig. 18(g), and some with nearly no lip motion like Fig. 18(a). We can see that our method has smaller $D_v(x)$ and thus successfully keeps temporal continuity in feature space regardless of motion magnitudes.

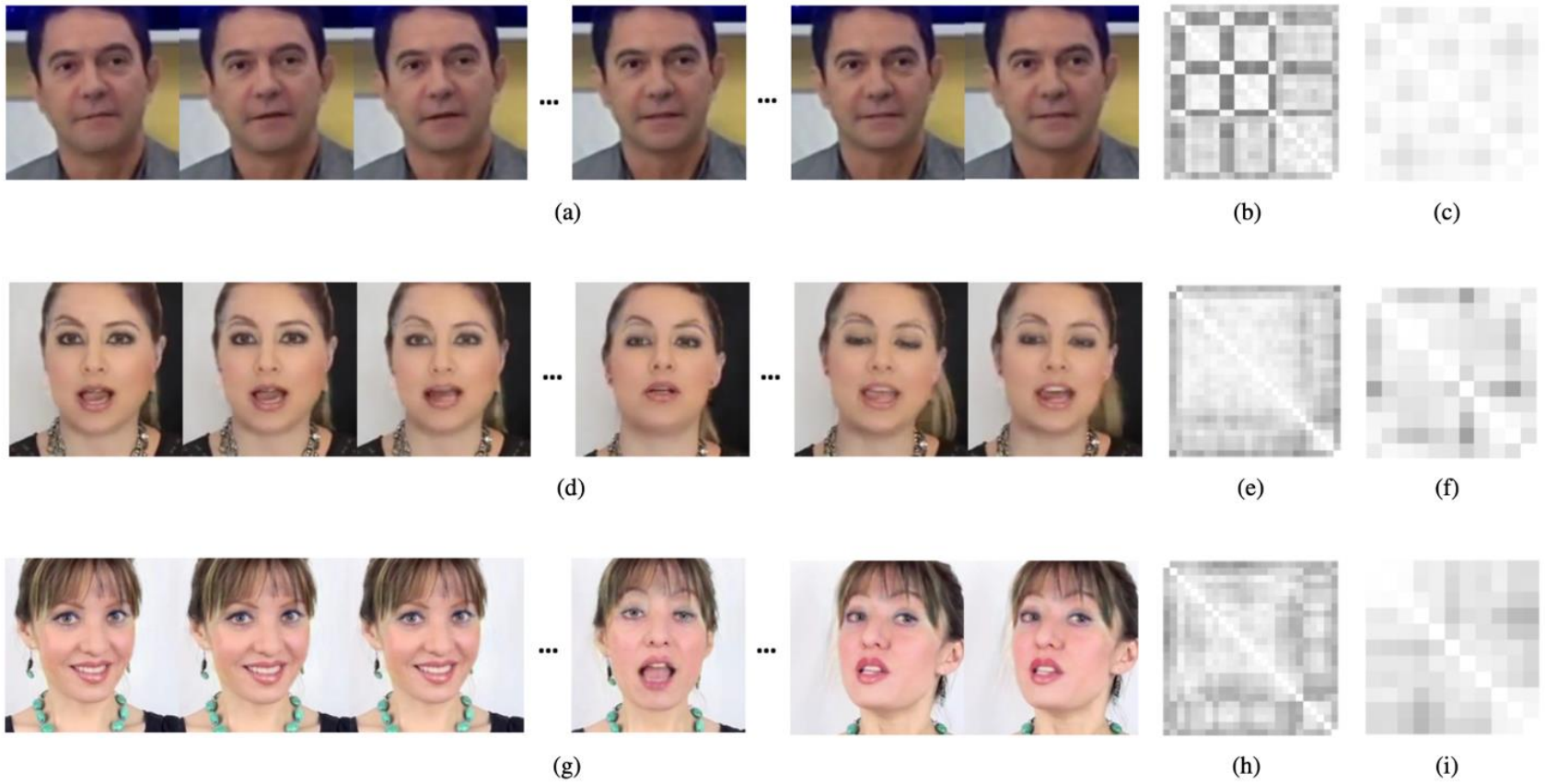

Fig. 18. Samples corrected by DTNet (ours): (a), (d), and (g): a sample of nearly no lip motion, a sample of big lip motion, a sample of big head pose change, respectively; (b), (e), and (h): similarity maps of corresponding samples in baseline, $D_v(x)$: 0.1781, 0.1205, 0.1468, respectively; (c), (f), and (i): similarity maps of corresponding samples in DTNet, $D_v(x)$: 0.0322, 0.0850, 0.0858, respectively.

5. Discussion

5.1 Irregularity of Lip Movements

As we proposed in Sec. 3.2, we regularize the deviation of motion magnitude variations in the loss function. It may raise some concerns that constraining feature vector deviation could lead to misclassification. We discuss and explain this concern in this section.

We argue that our model only regularizes feature vectors variations caused by motion magnitudes. And we do not rely on the motion magnitude regularities to classify. So it is unlikely to impair classification results after regularization. Here is the reason: The motion magnitude is a low-level feature, it may achieve good results for specific DeepFake generation methods. But it cannot generalize to novel methods as we discussed in Sec 2. And here is how we do not rely on motion magnitudes: we pretrain the model in Lipreading dataset, which enables the model to extract semantic information [6] and detect DeepFake based on the semantic information.

We also argue that regularizing feature vectors variations caused by motion magnitudes contributes to a better feature representation. Lipforensics [6] showed that semantic information is a discriminative and generalized feature to detect DeepFake. However, motion magnitudes impaired semantic information representation in current methods. For example, when lips conveyed the same word, lip motions with different motion magnitudes should have the same semantic information. But motion magnitudes confused the network model and it may mistakenly output that lips have different semantic information, which further impaired DeepFake detection. We regularize the motion magnitude's variations and thus get a more accurate semantic feature representation.

5.2 Limitations

Firstly, although our model reduces the reliance on pretraining data compared

to Lipforensics [6], we still required pretrain data. The pretraining data in lip reading is evitable. Because the DeepFake detection of our model and Lipforensics is based on the semantic information extraction on lip motions. The model needs to be pretrained in the lip-reading task first to make it feasible for the model to extract semantic information on lip motions. Without the pretraining, we lost the semantic meaning of our feature representation. We choose to keep the pretraining part the same as Lipforensics as the extraction of semantic meaning plays a vital role in boosting generalization of the model.

Secondly, rare samples with the extreme head pose (e.g. only half face appears) remain challenging for us as well as all current methods. The lack of training DeepFake samples with the extreme head pose may be a significant reason for the occurrence of this limitations.

Thirdly, the preprocessing stage to crop the mouth area is less efficient compared with the inference stage. We need to crop the mouth area of multiple frames and align the areas across the frames before input into the model.

Fourthly, model pretraining inevitably requires significant data and computational resources. Similar to other video processing tasks, our task is highly time-consuming and requires significant computational resources. Despite these demands, we utilize videos because the multiple frames provide crucial inter-frame inconsistencies, leading to better prediction performance compared to single images. During the pretraining phase, akin to LipForensics, our approach necessitates a large amount of data and computational resources to enable the model to learn high-level semantic information. This is vital for the subsequent training stages of the model. The pretraining phase was conducted using eight GeForce RTX 4080 Ti GPUs, while the subsequent training phase utilized two GeForce RTX 4080 Ti GPUs.

6. Conclusion and Future Work

6.1 Conclusion

We propose DTNet for DeepFake detection that aims to regularize feature vector learning of semantic information. DTNet is more robust to different motion magnitudes than previous methods, and successfully improves the model generalization ability to unseen DeepFake generation methods.

6.1 Future Work

Firstly, it would be promising to incorporate other types of high-level information. Our method shows the great potential of high-level information i.e., the semantic meaning of lip motions, to increase model generalization. Other types of high-level information, like identity and face expressions, that utilize the information of whole faces (our model only uses the mouth area), are worth exploring.

Secondly, it would be possible to pay more attention to samples with extreme pose in future work. Even dataset with various types of samples like DFDC has few samples with the extreme head pose. Current improvements of DeepFake datasets mainly focused on developing more sophisticated methods to generate more realistic samples. This limitation gives us another direction of developing DeepFake datasets in the future. Also, faces with the extreme head pose is a challenging topic in face recognition field. Some useful methods can be transferred into DeepFake detection field.

Thirdly, the number of frames is an important factor that influences the accuracy of the model, as it is discussed in many papers involved video processing. More frames of input generally give higher accuracy, but it does increase the computational burden. Although it is a common trick to balance the increase of the accuracy and the increase of the computational burden when fine-tuning the number of input frames, it is worthwhile to explore methods that can extract features in temporal level more efficiently.

Fourthly,

Bibliography

[1] Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial networks." Communications of the ACM 63, no. 11 (2020): 139-144.
[2] Korshunova, Iryna, Wenzhe Shi, Joni Dambre, and Lucas Theis. "Fast face-swap using convolutional neural networks." In Proceedings of the IEEE international conference on computer vision, pp. 3677-3685. 2017. [3] Zhu, Hao, et al. "Deep audio-visual learning: A survey." International Journal of Automation and Computing 18 (2021): 351-376.
[4] Pantserev, Konstantin A. "The malicious use of AI-based deepfake technology as the new threat to psychological security and political stability." Cyber defence in the age of AI, smart societies and augmented humanity (2020): 37-55.
[5] Westerlund, Mika. "The emergence of deepfake technology: A review." Technology innovation management review 9, no. 11 (2019).
[6] Haliassos, Alexandros, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic. "Lips don't lie: A generalisable and robust approach to face forgery detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5039-5049. 2021.
[7] Faceswap," https://github.com/MarekKowalski/FaceSwap, Accessed: 2022-12-01.
[8] Kim, Hyeongwoo, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. "Deep video portraits." ACM Transactions on Graphics (TOG) 37, no. 4 (2018): 1-14.
[9] Thies, Justus, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. "Face2face: Real-time face capture and reenactment of rgb videos." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2387-2395. 2016.

[10] Lample, Guillaume, Neil Zeghidour, Nicolas Usunier, Antoine Bordes, Ludovic Denoyer, and Marc'Aurelio Ranzato. "Fader networks: Manipulating images by sliding attributes." Advances in neural information processing systems 30 (2017).

[11] Wu, Po-Wei, Yu-Jing Lin, Che-Han Chang, Edward Y. Chang, and Shih-Wei Liao. "Relgan: Multi-domain image-to-image translation via relative attributes." In Proceedings of the IEEE/CVF international conference on computer vision, pp. 5914-5922. 2019.

[12] Choi, Yunjey, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. "Stargan v2: Diverse image synthesis for multiple domains." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 8188-8197. 2020.

[13] Tolosana, Ruben, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. "Deepfakes and beyond: A survey of face manipulation and fake detection." Information Fusion 64 (2020): 131-148.

[14] Li, Yuezun, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. "Celeb-df: A large-scale challenging dataset for deepfake forensics." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3207-3216. 2020.

[15] Rossler, Andreas, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. "Faceforensics++: Learning to detect manipulated facial images." In Proceedings of the IEEE/CVF international conference on computer vision, pp. 1-11. 2019.

[16] Thies, Justus, Michael Zollhöfer, and Matthias Nießner. "Deferred neural rendering: Image synthesis using neural textures." Acm Transactions on Graphics (TOG) 38, no. 4 (2019): 1-12.

[17] Faceswap app," https://github.com/deepfakes/faceswap, Accessed: 2023-02-22

[18] Li, Lingzhi, Jianmin Bao, Hao Yang, Dong Chen, and Fang Wen. "Advancing high fidelity identity swapping for forgery detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5074-5083. 2020.

[19] Dolhansky, Brian, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. "The deepfake detection challenge

(dfdc) dataset." arXiv preprint arXiv:2006.07397 (2020).
[20] Huang, Dong, and Fernando De La Torre. "Facial action transfer with personalized bilinear regression." In Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part II 12, pp. 144-158. Springer Berlin Heidelberg, 2012.
[21] Abdal, Rameen, Yipeng Qin, and Peter Wonka. "Image2stylegan: How to embed images into the stylegan latent space?." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4432-4441. 2019.
[22] Nirkin, Yuval, Yosi Keller, and Tal Hassner. "Fsgan: Subject agnostic face swapping and reenactment." In Proceedings of the IEEE/CVF international conference on computer vision, pp. 7184-7193. 2019.
[23] Li, Lingzhi, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. "Face x-ray for more general face forgery detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5001-5010. 2020.
[24] Frank, Joel, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. "Leveraging frequency analysis for deep fake image recognition." In International conference on machine learning, pp. 3247-3258. PMLR, 2020.
[25] Li, Jiaming, Hongtao Xie, Jiahong Li, Zhongyuan Wang, and Yongdong Zhang. "Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 6458-6467. 2021.
[26] Luo, Yuchen, Yong Zhang, Junchi Yan, and Wei Liu. "Generalizing face forgery detection with high-frequency features." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 16317-16326. 2021.
[27] Amerini, Irene, Leonardo Galteri, Roberto Caldelli, and Alberto Del Bimbo. "Deepfake video detection through optical flow based cnn." In Proceedings of the IEEE/CVF international conference on computer vision workshops, pp. 0-0. 2019.
[28] Sabir, Ekraam, Jiaxin Cheng, Ayush Jaiswal, Wael AbdAlmageed, Iacopo Masi, and Prem Natarajan. "Recurrent convolutional strategies for face manipulation detection in videos." Interfaces (GUI) 3, no. 1 (2019): 80-87.

[29] Zheng, Yinglin, Jianmin Bao, Dong Chen, Ming Zeng, and Fang Wen. "Exploring temporal coherence for more general video face forgery detection." In Proceedings of the IEEE/CVF international conference on computer vision, pp. 15044-15054. 2021.
[30] Masi, Iacopo, Aditya Killekar, Royston Marian Mascarenhas, Shenoy Pratik Gurudatt, and Wael AbdAlmageed. "Two-branch recurrent network for isolating deepfakes in videos." In Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16, pp. 667-684. Springer International Publishing, 2020.
[31] Chen, Liang, Yong Zhang, Yibing Song, Lingqiao Liu, and Jue Wang. "Self-supervised learning of adversarial example: Towards good generalizations for deepfake detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 18710-18719. 2022.
[32] Chen, Shen, Taiping Yao, Yang Chen, Shouhong Ding, Jilin Li, and Rongrong Ji. "Local relation learning for face forgery detection." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 2, pp. 1081-1088. 2021.
[33] Zhao, Tianchen, Xiang Xu, Mingze Xu, Hui Ding, Yuanjun Xiong, and Wei Xia. "Learning self-consistency for deepfake detection." In Proceedings of the IEEE/CVF international conference on computer vision, pp. 15023-15033. 2021.
[34] Zhou, Yipin, and Ser-Nam Lim. "Joint audio-visual deepfake detection." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 14800-14809. 2021.
[35] Gu, Yewei, Xianfeng Zhao, Chen Gong, and Xiaowei Yi. "Deepfake Video Detection Using Audio-Visual Consistency." In Digital Forensics and Watermarking: 19th International Workshop, IWDW 2020, Melbourne, VIC, Australia, November 25–27, 2020, Revised Selected Papers 19, pp. 168-180. Springer International Publishing, 2021.
[36] Zhao, Hanqing, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu. "Multi-attentional deepfake detection." In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 2185-2194. 2021.
[37] Wang, Junke, Zuxuan Wu, Wenhao Ouyang, Xintong Han, Jingjing Chen,

Yu-Gang Jiang, and Ser-Nam Li. "M2tr: Multi-modal multi-scale transformers for deepfake detection." In Proceedings of the 2022 International Conference on Multimedia Retrieval, pp. 615-623. 2022.
[38] Dong, Xiaoyi, Jianmin Bao, Dongdong Chen, Ting Zhang, Weiming Zhang, Nenghai Yu, Dong Chen, Fang Wen, and Baining Guo. "Protecting celebrities from deepfake with identity consistency transformer." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 9468-9478. 2022.
[39] Cozzolino, Davide, Andreas Rössler, Justus Thies, Matthias Nießner, and Luisa Verdoliva. "Id-reveal: Identity-aware deepfake video detection." In Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 15108-15117. 2021.
[40] Agarwal, Shruti, Hany Farid, Tarek El-Gaaly, and Ser-Nam Lim. "Detecting deep-fake videos from appearance and behavior." In 2020 IEEE international workshop on information forensics and security (WIFS), pp. 1-6. IEEE, 2020.
[41] Chung, Joon Son, and Andrew Zisserman. "Lip reading in the wild." In Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part II 13, pp. 87-103. Springer International Publishing, 2017.
[42] Martinez, Brais, Pingchuan Ma, Stavros Petridis, and Maja Pantic. "Lipreading using temporal convolutional networks." In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 6319-6323. IEEE, 2020.
[43] Nguyen, Huy H., Fuming Fang, Junichi Yamagishi, and Isao Echizen. "Multi-task learning for detecting and segmenting manipulated facial images and videos." In 2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS), pp. 1-8. IEEE, 2019.