An efficient approach for Deepfake Detection Employing Micro-Expressions with a Hierarchical Transformer Network

by

Md. Ismail
21141012
Shah Md. Labib Shadik
20301034
Md. Sajid Ullah Sohan
20301129
MD. Abir Anam Bhuiyan
20101058
Lamia Khan Shoily
20301085

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
January 2025

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Md. Ismail	Shah Md. Labib Shadik
21141012	20301034
MD. Abir Anam Bhuiyan	Md. Sajid Ullah Sohan
20101058	20301129

Lamia Khan Shoily

20301085

Approval

The thesis/project titled "An efficient approach for Deepfake Detection Employing Micro-Expressions with a Hierarchical Transformer Network" submitted by

1. Md. Ismail(21141012)
2. Md. Sajid Ullah Sohan (20301129)
3. Shah Md. Labib Shadik (20301034)
4. MD. Abir Anam Bhuiyan (20101058)
5. Lamia Khan Shoily (20301085)

Of Spring, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 24, 2025.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
PhD Professor
Department of Computer Science & Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

The widespread use of deepfake technology makes it difficult to tell the difference between modified and real digital content, as deepfakes increasingly use digitally altered faces or bodies to mimic others for malicious purposes or to spread false information. With artificial intelligence technology improving, deepfakes are becoming more realistic and challenging to detect using conventional techniques that focus on surface-level visual cues like pixel irregularities and visual artifacts, which may soon be insufficient due to the rapid advancement of deepfake generation techniques. This study adapts the Hierarchical Transformer Network (HTNet), originally developed for recognizing micro-expressions, to detect deepfake videos. The HTNet, known for its ability to identify subtle involuntary facial movements, is modified to discern the synthetic traces typically undetectable by traditional methods. The model is particularly attentive to critical facial areas such as the eyes and lips, which often display signs of manipulation. Optical flow is utilized to track subtle motion between frames in both real and deepfake content. The performance of the retrained HTNet model is rigorously evaluated using standard micro-expression benchmarks, including SAMM, SMIC, CASME II, and CASME III, and its effectiveness in deepfake detection is thoroughly assessed using the FaceForensics++ dataset. Tests on deepfake and micro-expression datasets demonstrate the model's capability in identifying fake videos, showcasing HTNet as an effective advanced model for tackling the growing problem of deepfakes.

Keywords: Micro-expression; Deepfake; Optical flow; ;HTNet; FaceForensic++; CASME II; SAMM; SMIC

Dedication

All thanks to Almighty Allah, the most magnificent; without His will, this task wouldn't be completed.

It is our great pleasure to see that each member has shown the utmost dedication to our work and all blessings to them.

We shall perpetually be grateful to our parents for their steadfast confidence in our abilities and assistance, as well as their collaboration, which inspired us to achieve scholastic success. Their heartfelt prayers and uplifting remarks contributed to our continued growth and maturation into the individuals we are today. As we embark on the subsequent phase of our existence, we recognise and appreciate the immense worth of their selflessness. We express profound gratitude for their steadfast support.

Acknowledgement

Allah is the greatest planner. His favour was important to let us finish our Final thesis. His guidance provided us with direction, strength, and knowledge to overcome obstacles along the way.

We have gotten constant support from our supervisor, Dr. Md. Ashraful Alam, sir. His supervision has made our academic journey more smooth and helpful. We want to thank you sir from the bottom of our hearts. He is not only our great supervisor but also a mentor and an inspiration.

The knowledge, constructive criticism, and support that Dr. Md. Ashraful Alam sir provided to us have been valuable in helping us shape our study. It also helped us to improve our knowledge of the field. We are really grateful to have had the chance to work under his direction. His commitment to academic quality has been clear in every sector.

We could not do it without recognising Dr. Md. Ashraful Alam sir's continuous availability and desire to share his depth of knowledge. His assistance helped us a lot. The understanding we have received from him will influence our future endeavours.

We are very honoured to have Dr. Md. Ashraful Alam sir for his guidance and significant influence on our academic and personal well-being.

We would also like to thank the founders of CASME II, SAMM, SMIC datasets for providing us the datasets on request. Their datasets are very important resource for completing our research.

Table of Contents

Declaration i

Approval ii

Abstract iii

Dedication iv

Acknowledgment v

Table of Contents vi

List of Figures ix

1 Introduction 1
- 1.1 Problem Statement . . . 2
- 1.2 Description of the problem . . . 2
- 1.3 Research Objective . . . 3

2 Literature Review 4
- 2.1 Introduction to Deepfake Technology and Its Challenges: . . . 4
- 2.2 Challenges in Deepfake Detection: . . . 5
- 2.3 Advancements in Deepfake Generation: . . . 5
- 2.4 Detection Techniques and Methodologies: . . . 6
- 2.5 Adversarial Attacks and Datasets: . . . 7
- 2.6 Facial Landmark and Motion-Based Deepfake detection . . . 8
- 2.7 Motion-Driven Approaches for Micro- Expression Identification and Deepfake Detection . . . 9
- 2.8 Hierarchical Transformer Network (HTNet) for Micro-Expression Recognition . . . 10
  - 2.8.1 Optical Flow and Kinematic Analysis . . . 10
  - 2.8.2 Hierarchical Attention Mechanisms . . . 11
  - 2.8.3 Rationale for Employing Micro-Expressions in Deepfake Identification . . . 11
  - 2.8.4 Potential Application to Deepfake Detection . . . 12

3 Description of model 13
- 3.1 Introduction . . . 13
- 3.2 HTNet Architecture . . . 13
  - 3.2.1 LayerNorm . . . 14

3.2.2 PreNorm . . . 14
3.2.3 FeedForward Layer . . . 15
3.2.4 Attention Layer . . . 15
3.2.5 Aggregate Layer . . . 16
3.2.6 HTNet Layer . . . 16
3.2.7 MLP Head . . . 16
3.3 Optical Map Extraction . . . 16
3.4 Transformer Layer: . . . 18
3.5 Block Aggregation . . . 19

4 Dataset **22**
4.1 Data Description . . . 22
4.1.1 FaceForensics++ . . . 22
4.1.2 Microexpression Dataset . . . 22
4.2 Data Preprocessing . . . 23
4.2.1 Facial Action Coding System and Action Units . . . 23
4.2.2 Onset,Apex,Offset . . . 24
4.2.3 Optical Flow . . . 25
4.2.4 Gunner-Farnebeck for Micro-expression . . . 25
4.2.5 RAFT . . . 26
4.3 TVL1 Optical Flow . . . 27
4.4 Data Analysis . . . 28
4.4.1 Mean Optical Flow Magnitude (MOFM) . . . 28
4.4.2 Optical Flow Variance . . . 28
4.4.3 Facial Action Unit (FAU) Correlation . . . 28

5 Methodology **30**
5.1 Methodology . . . 30
5.1.1 Dataset Preparation and Preprocessing . . . 30
5.1.2 Preprocessing . . . 31
5.1.3 Deepfake Detection via HTNet Adaptation . . . 31
5.2 Training Strategy . . . 33
5.2.1 Transfer Learning/From-scratch learning . . . 33
5.2.2 Fine-Tuning . . . 34
5.3 Model Training and Evaluation . . . 34
5.3.1 Loss Functions and Optimizers . . . 34
5.3.2 Validation and Testing . . . 34
5.3.3 Transformer-Based Deep Feature Integration . . . 36
5.4 Robust Classification and Prediction . . . 37

6 Result **39**

7 Comparative Analysis **46**
7.1 Comparative Performance Analysis of Different Approaches for Deepfake Detection . . . 46
7.1.1 Analysis . . . 47

8 Observation and Discussion **49**

9 Conclusion **52**

Bibliography **55**

List of Figures

3.1 Model Architecture 18
3.2 Transfer Learning Pipeline 20

4.1 Optical flow generation 26
4.2 RAFT optical flow 27

6.1 Convergence curve of HTNet during training. 39
6.2 Accuracy Curve of HTNet 40
6.3 F1 Score Curve of HTNet 41
6.4 Precision Curve of HTNet 42
6.5 Recall Curve of HTNet 43
6.6 Recall Curve of HTNet 44
6.7 Confusion Matrix of HTNet Model 45

Chapter 1

Introduction

Deepfake detection has become an essential area of research due to the increasing misuse of AI-generated content to create realistic yet fabricated videos. One promising approach for detecting deepfakes involves analyzing micro-expressions — subtle facial muscle movements that occur involuntarily and last only for a fraction of a second. These micro-expressions are difficult to replicate authentically in deepfake videos, making them a reliable indicator for distinguishing real from fake content. In our research, we explored the use of micro-expression analysis for deepfake detection by leveraging the Hierarchical Transformer Network (HTNet) as proposed in the reference paper. HTNet effectively captures both local and global facial features by focusing on four key facial regions: the left eye, right eye, left lip, and right lip areas. This allows for precise identification of subtle muscle movements crucial for micro-expression recognition. We adopted two approaches for deepfake detection using micro-expression analysis: one utilizing transfer learning and the other without it. Transfer learning involves using a pre-trained model, allowing us to transfer knowledge from the micro-expression recognition task to deepfake detection. This method helps overcome the challenge of limited labeled deepfake datasets by leveraging learned facial feature representations. On the other hand, the non-transfer learning approach trains the model from scratch, focusing solely on the deepfake dataset without prior knowledge. Comparing these methods provides insights into the effectiveness of pre-learned micro-expression features versus learning directly from deepfake data. Our research aims to highlight the importance of micro-expressions in uncovering deepfakes and assess whether transfer learning enhances detection accuracy. By combining micro-expression analysis and deep learning techniques, we take a step toward building more robust and reliable deepfake detection systems.

Why HTNet model:

The Htnet model was selected because it performs better than the other models, as explained in the model description. Its hierarchical transformer architecture captures the grained features on a local level to coarse-grained features on a global level. This model separates the optical fluxes into several zones of interest and combines them, which enables the model to better capture the delicate facial dynamics than any other model that currently exists for microexpression. As a result, it is better at capturing microexpressions, which is the final goal, because capturing microex-

pressions better would produce better results for deepfake detection since the end goal is to identify the deepfake using microexpressions.

1.1 Problem Statement

As deepfake technology continues to improve, there is a need for a new method of detecting deepfakes that makes use of microexpressions. Deepfake technology is not yet able to effectively recreate microexpressions, and there is currently technology available that can skip surface-level indications, making this a very effective technique.

1.2 Description of the problem

With today's advanced technology, creating fake videos of someone to damage their reputation has become easier than ever. This has turned into a serious global problem. Deepfake technology has grown as a significant challenge in digital media In recent years, leveraging deep learning models to create highly realistic yet entirely fabricated videos and images. Deepfake content has been used in various malicious ways, such as political misinformation, identity fraud, and non-consensual media manipulation, posing serious ethical and security concerns [11]. While significant progress has been made in deepfake detection, many existing approaches struggle with generalization across datasets and remain vulnerable to adversarial attacks [12]. When compared to authentic video, the existing algorithms do provide a high level of accuracy in identifying deepfakes[11]. The problem is that, just as artificial intelligence technology has advanced in recent years, so too has deepfake technology, which can now produce high-quality, realistic deepfake content with human-like facial expressions that make it very difficult for the general public to distinguish between the two. Additionally, existing technology that only detects deepfakes using surface-level indicators like pixel intensities, facial anomalies, facial synchronizing, etc. is being surpassed as deepfake technology advances [7]. Thus, a new method is required for deepfake detection that does not rely on traditional methods like surface level cues, which are used by some traditional models like LSTM, RNN, CNN, etc. Instead, it considers new dilemmas and focuses on developing new methods and models for deepfake detection.

Microexpressions, which are brief, involuntary facial muscle movements occurring within 0.04 to 0.2 seconds, offer a promising new avenue for deepfake detection [4]. Unlike macroexpressions, microexpressions are difficult to control and replicate artificially, making them a strong behavioral biometric for identifying manipulated content. Recent research suggests that deepfake generation techniques often fail to reproduce authentic microexpressions due to limitations in facial reenactment models and GAN-based synthesis processes [10]. This opens the door for using microexpression-based features as a new forensic cue for deepfake detection.

1.3 Research Objective

This paper proposes a novel deepfake detection approach by leveraging microexpression analysis through both transfer learning and non-transfer learning approaches. Transfer learning enables the adaptation of pre-trained weights, particularly those developed for microexpression recognition, to the domain of deepfake detection. Instead of training from scratch, transfer learning allows the reuse of learned spatiotemporal representations, significantly reducing the computational burden while improving detection accuracy [19].
In contrast, the non-transfer learning approach involves training models directly on deepfake datasets without relying on pre-learned representations. For this, we use the FaceForensics++ dataset, allowing the model to learn microexpression patterns solely from deepfake data. This approach helps us understand the effectiveness of detecting deepfakes without any external knowledge transfer.
Existing deepfake detection models primarily rely on spatial inconsistencies in face images or deep-learning-based classifiers, but they often lack a fine-grained temporal understanding of facial dynamics [16]. By incorporating microexpression recognition models, our approach enhances deepfake detection by identifying discrepancies in facial muscle movements across video frames.
Our research will build upon established microexpression datasets like CASME II and SAMM and transfer their learned feature representations to deepfake datasets like FaceForensics++. The integration of HTNet models, which have demonstrated strong performance in microexpression recognition, will be explored to capture subtle, transient facial cues that deepfake generators struggle to replicate accurately [19]. By leveraging these insights, we aim to improve deepfake detection robustness against diverse manipulation techniques and compressed video formats.
This thesis will focus on the following key research questions:

- Can microexpression features enhance the accuracy of deepfake detection compared to traditional CNN-based methods?
- How effective is transfer learning in adapting microexpression models to deepfake datasets?
- How effective is adapting microexpression models to deepfake datasets without transfer learning?

By addressing these questions, this research aims to contribute a new dimension to deepfake detection, demonstrating the potential of microexpression-based forensic analysis as a reliable countermeasure against synthetic media manipulations. Through comprehensive experimentation and comparison of transfer learning and non-transfer learning approaches, we strive to build a more robust, adaptive, and accurate deepfake detection framework.

Chapter 2

Literature Review

The rapid progress in generative models has resulted in an unparalleled increase in deepfake production, presenting considerable ethical, social, and security dilemmas. As deepfakes enhance in quality, efficient detection methods must concentrate on recognizing nuanced and involuntary face movements, including micro-expressions. This literature review analyses advancements in deepfake detection techniques. It emphasizes the integration of optical flow, facial expression dynamics, motion analysis, and our primary paper: the utilization of micro-expressions for deepfake detection, as these methodologies offer a reliable means of detecting subtle discrepancies in synthetic media that are difficult for deepfake technologies to replicate accurately.

2.1 Introduction to Deepfake Technology and Its Challenges:

The study paper titled "Deepfake Technology in Corporate Cybersecurity: Emerging Threats and Defence Mechanisms" authored by Dr. Felix Hernandez examines the advancing dangers posed by deepfake technology in corporate cybersecurity and emphasises solutions for mitigation. The paper examines the progress in deep learning and generative adversarial networks (GANs), which facilitate the production of very realistic synthetic media, hence presenting risks such as identity theft, fraud, and disinformation. It underscores the imperative of integrating automated detection methodologies, such as facial landmark analysis, lip synchronisation detection, and micro-expression analysis, with human supervision for a resilient defence. The report emphasises the necessity of adopting zero-trust models and multi-factor authentication as preventive strategies. The article identifies possible opportunities presented by deepfake technology, provided it is utilised properly, while also addressing the associated problems. The results emphasise the immediate necessity for enterprises to prioritize awareness, readiness, and investment in sophisticated cybersecurity systems.[2]

Deepfake technology, driven by GANs and autoencoders, enables authentic face swapping and reenactment, yet presents societal hazards due to potential misuse. The study conducted by Peipeng Yu et al.[15] classifies detection methodologies into general network-based approaches, temporal consistency techniques, and artifact-based detection methods. CNN-based models such as XceptionNet demon-

strate efficacy; nevertheless, they frequently experience overfitting, which constrains their generalisation across datasets. Temporal methodologies, including CNN-RNN frameworks, proficiently identify motion discrepancies resulting from frame-by-frame synthesis, akin to the motion-centric approaches employed in micro-expression analysis. Artifact-based methods, which emphasise the integration of boundaries and discrepancies in head posture, show potential but face challenges due to advancing manipulation capabilities. The paper emphasizes the value of multitask learning and resilient training mechanisms for promoting detection generalisation and competence. The merging of motion dynamics and temporal examination, employed by micro-expression recognition, is an effective foundation in handling subtle deceit in deepfake detection [26] [25]

2.2 Challenges in Deepfake Detection:

— In "Limits of Deepfake Detection: A Robust Estimation Viewpoint,"[1] Sakshi Agarwal and Lav R. Varshney analyze the challenge posed by deepfake detection as Generative Adversarial Networks become more realistic. The contributions of the research consist of framing the detection problem as a hypothesis testing matter anddemonstrating by means of rigorous statistical frameworks that in conditions where GAN-generated distributions converge to the distribution of real data, detection error increases exponentially, particularly in the case of high resolution images. With increasingly sophisticated GANs like StyleGAN inthe mix, more traditional techniques relying on visual artefacts or temporal inconsistencies — for example, by detecting warping or blinking anomalies — become less effective. Our researchhighlights the limitations of existing approaches and advocates for adaptive, generalised detection frameworks in tackling increasingly realistic deepfakes.
[20]
-In "Why DoFacial Deepfake Detectors Fail?" by Binh Le et al.,[21] the authors highlight the shortcomings of current deepfake detection technologies, including preprocessing differences, restriction to certain datasets,and insufficient generalisation to unseen deepfakes. They show that scaling during preprocessing alters vital features, generative noise can ruin detection, and shifts in datasets strikingly lower performance as shown by a slide from 98% accuracy with models pretrained on FaceForensics++ to roughly 52% accuracy onCeleb-DF. The study emphasizes the importance of robust andgeneralized detection systems to keep pace with evolving deepfake techniques. [24] [7] [4]

2.3 Advancements in Deepfake Generation:

- In "Latent Flow Diffusion for Deepfake Video Generation," Aashish Chandra K et al. propose a novel deepfake generation approach with latent flow diffusion (LFD) to improve temporal consistency and spatial accuracy in generated movies. The model has an optical flow predictor that learns motion dynamics across frames in the latent space, which offers temporally consistent deepfake movies. The approach employs a double-transformer encoding block to align the target image's spatial features with the source video's temporal dynamics and employs a 3D denoising diffusion model

to render video. The paper emphasizes the importance of optical flow in preserving the original video's motion continuity since it warps valuable temporal information. Optical flow enables the model to integrate spatial and temporal characteristics, addressing issues with creating realistic facial expressions and movement in deepfake movies. Benchmarks such as FaceForensics++ confirm the enhanced effectiveness of the model in producing quality deepfake movies over baseline practices.[23] [17]

- "A Systematic Literature Review on the Effectiveness of Deepfake Detection Techniques" by Laura Stroebel et al, reviews the techniques of deepfake detection systematically and reports the superiority of deep learning (DL) over traditional machine learning in its feature extraction ability and detection rate. While these are steps in the right direction, the study reports that traditional deep learning models are diminishing in effectiveness when it comes to handling diverse real-world deepfakes. Overreliance on biased data and limited diversity compromises the generalizability of models, which do not generalize to novel data. The survey mentions future directions, including the incorporation of audio-visual modalities, enhancing temporal consistency modeling, and mitigating dataset biases to construct robust and scalable detection mechanisms and "Deepfake Video Detection: Challenges and Opportunities" by Achhardeep Kaur et al. discusses prominent challenges in deepfake video detection, including computational complexity, overwhelming dependence on data-hungry models, and poor generalization performance across datasets. The paper recognizes the difference between image and video deepfake detection, where detection in videos is more complex and includes temporal consistency analysis, with increased complexity and computational overhead. The paper further recognizes that current detection approaches are vulnerable to overfitting certain datasets, with low performance reported in real-world settings. The authors propose hybrid approaches, with frequency domain analysis incorporated in spatial-temporal modeling, for detecting reliability and adversarial robustness and against advanced manipulations.[2] [27]

2.4 Detection Techniques and Methodologies:

- Ruben Tolosana et al.'s paper "DeepFakes and Beyond: A Survey of Face Manipulation and Fake Detection"classifies facial manipulation methods into four broad categories: face synthesis from scratch, identity exchange, attribute modification, and expression exchange. It explains how the recent developments in manipulation algorithms, fueled by GAN-based methods such as StyleGAN, are outpacing detection systems. Detection methods being investigated include convolutional neural networks, attention mechanism, and temporal analysis with special focus on motion-based features such as optical flow to detect anomalies in tampered videos. Simultaneously, micro-expression recognition research, such as that by Xiaobai Li et al. "A survey of micro-expression recognition"[16], points to subtle facial movements and temporal dynamics to detect subtle expressions. Both areas emphasize the need to examine high-resolution spatial and temporal characteristics to enhance detection performance. The applications of attention mechanisms and motion-based analysis in both areas indicate a converging trend where the findings of micro-expression research can enhance the flexibility and generalisability of manipulation detection systems, especially for capturing subtle and evasive manipulations.[11] [20]

2.5 Adversarial Attacks and Datasets:

-The article "FaceForensics++: Learning to Detect Manipulated Facial Images" by Andreas Rössler et al. presents a comprehensive dataset and benchmark for identifying face alterations in photographs and movies. The dataset comprises approximately 1.8 million altered photos generated by four techniques: DeepFakes, FaceSwap, Face2Face, and NeuralTextures, across diverse compression settings to emulate real-world conditions. The research assesses various advanced forgery detection techniques, concluding that deep learning models such as XceptionNet surpass human capabilities and other methods in identifying manipulations, even with significant compression applied. This dataset and benchmark seek to standardise assessments and enhance the resilience of forgery detection techniques under actual settings.[13] [14]

-The article "Making DeepFakes More Spurious: Evading Deep Face Forgery Detection via Trace Removal Attack" by Chi Liu et al. presents a detector-agnostic assault technique known as the Trace Removal Attack, aimed at circumventing deepfake detection mechanisms.[5] The method aims to eradicate intrinsic identifiable markers in deepfake images, including spatial anomalies, spectral discrepancies, and noise fingerprints, which detectors commonly utilise to distinguish authentic from fabricated images. The Trace Removal Network uses a generator and numerous discriminators to concurrently eliminate traces across diverse domains, enhancing deepfakes to closely mimic authentic images while maintaining visual integrity. The suggested method markedly diminishes detection accuracy across six advanced detectors, including those with strong defences, resulting in universal transferability to unidentified detectors. The paper illustrates that the existing dependence on superficial artefact detection is inadequate, emphasising the necessity for sophisticated detection systems to combat such attacks.[22] -In the article "AVA: Inconspicuous Attribute Variation-Based Adversarial Attack Bypassing DeepFake Detection" by Xiangtao Meng et al., the authors introduce a new adversarial assault technique termed Attribute-Variation-Based Adversarial Attack (AVA). This technique implements subtle modifications to prominent features, including mouth position, hairdo, and skin tone, within the latent space of DeepFake photos to evade detection algorithms. In contrast to conventional pixel-level attacks, AVA alters semantic properties through Gaussian priors and a semantic discriminator to preserve naturalness and remain imperceptible to human observers. The AVA assault was evaluated on nine advanced DeepFake detection models, including commercial systems such as Tencent and Baidu, successfully evading these detectors with a rate above 95%. **<empty citation>** The approach exceeds current pixel-level adversarial attacks in efficacy and visual quality. The article points out the susceptibility of spatial and frequency-domain artefact-based detection methods, illustrating the necessity for more powerful methods to counter attribute-level adversarial attacks. A human trial demonstrated that AVA-altered images are almost indistinguishable from genuine images, raising serious privacy and security issues.[27] [9] [10]

2.6 Facial Landmark and Motion-Based Deepfake detection

"Face Forgery Video Detection Based on Expression Key Sequences" by Yameng Tu et al. introduces a novel approach to deepfake detection by utilizing facial expression essential sequences rather than entire video sequences, significantly improving detection accuracy and computational efficiency. The method introduced here leverages optical flow to obtain dynamic facial expression sequences. It records movement between consecutive frames to identify minute inconsistencies. Such precious sequences are fed into a visual transformer-based spatio-temporal dual-branch inconsistency detection network. The spatial branch engages focal self-attention to extract cross-correlation features across expression areas. While the temporal branch uses optical flow disparities to identify inconsistencies in face movement.Optical flow plays a critical role by accurately extracting and highlighting dynamic changes in facial expressions, which are harder to synthesize without leaving forgery traces. It enables the detection model to focus on motion dynamics, reducing computational redundancy and improving reliability. The results that came from the Experiments on the FaceForensics++ dataset show that this method decreases training and detection time by 75% and 90%, respectively while outperforming state-of-the-art techniques in accuracy. The study underscores the effectiveness of optical flow in capturing subtle temporal inconsistencies, which are pivotal for detecting high-quality forgeries.[28]

-The article, "Deepfake Video Detection through Optical Flow Based CNN" by Irene Amerini, presents a new approach for detecting deepfake films by examining inter-frame motion discrepancies via optical flow analysis. This method, in contrast to conventional frame-based approaches, employs motion vector fields produced by optical flow (via PWC-Net) to capture the temporal irregularities typical of synthetic videos. The flow fields are utilised as input for convolutional neural networks (VGG16 and ResNet50) to perform binary classification (fake or real). Initial studies on the FaceForensics++ dataset demonstrate encouraging outcomes, especially in detecting manipulations such as Face2Face, achieving accuracies over 81%.. This method recognizes the promise of applying temporal dynamics to deepfake detection and proposes avenues for future work, such as experimentation on larger data sets and integration of frame-based approaches for improved performance.[22]

- The article, "FAMM: Facial Muscle Motions for Detecting Compressed Deepfake Videos Over Social Networks" by Xin Liao, proposes the FAMM model that uses facial muscle motion features derived from geometric face landmark analysis for detecting compressed deepfake videos. In contrast to conventional techniques that concentrate on manipulation artefacts, FAMM detects false facial movements resulting from temporal discrepancies in counterfeit movies. The framework employs GRU and SVM classifiers to analyse inter-frame changes and temporal sequences, utilising Dempster-Shafer theory for predictive fusion. Comprehensive trials on the FaceForensics++ datasets illustrate FAMM's resilience and superior efficacy in identifying compressed movies, even inside real-world social media contexts.[8]

- Hanqing Zhao has proposed a new method for detecting deepfake movies as a detailed classification task in "Multi-Attentional Deepfake Detection". The model applies multi spatial attention heads to pay attention to specific local areas of face features, a texture enhancement block to magnify minor artefacts in shallow features, and a bilinear attention pooling module to integrate fine-texture details with abstract facial representations. The authors suggest a loss of regional autonomy to improve the training of such a multi-attentional network in a manner that attention heads focus on disparate regions, and an attention-guided data augmentation strategy to improve robustness. Benchmark tests on FaceForensics++, Celeb-DF, and DFDC demonstrate the superior performance of the method, particularly for combatting subtle and localised artefacts in deepfake detection.[18]

2.7 Motion-Driven Approaches for Micro- Expression Identification and Deepfake Detection

-In the article "Main Directional Mean Optical Flow (MDMO) for Micro-Expression Recognition," Liu et al. (2015) introduced the MDMO technique, which utilises optical flow to detect nuanced motion patterns in micro-expression identification. The method segments the face into 36 regions of interest (ROIs) informed by facial action units and calculates normalised motion statistics based on the predominant optical flow direction in each region. This procedure significantly reduces computational complexities while supporting the capacity to recognise subtle changes, rendering it extremely efficient. The evaluation of the SMIC, CASME II and CASME databases shows that MDMO exceeded the conventional systems like LBP-TOP, indicating its toughness in managing difficult situations, including variations in lighting and minor head gestures. The procedure's significance on motion dynamics deeply correlates with recognising inconsistencies in deepfakes, where facial movement's minute alterations must be detected.[18]

-In the article "Shallow Triple Stream Three-Dimensional CNN (STSTNet) for Micro-Expression Recognition," Liong et al. (2019) addressed STSTNet, a simplified architecture designed for micro-expression recognition. This technique includes optical flow, optical strain and apex frame detection to recognise and test slight facial deformity. The three-pathway system examines horizontal and vertical optical flow and optical deformed components while capturing motion information efficiently. Despite its superficial architecture, STSNEet achieves excellent performance on benchmark datasets which include CASME II, SMIC and SAMM, revealing notable accuracy and recall metrics. The implementation of optical strain, which signifies the scale of facial distortions also renders STSNet mainly functional for detection of deepfake, where recognising subtle and complex facial changes is a must.[6]

-In the article "Fuzzy Histogram of Optical Flow Orientations (FHOFO) for Micro-Expression Recognition," Happy and Routray (2017) introduced FHOFO to handle the difficulties related with catching nuanced motion dynamics in micro-expressions. FHOFO generalizes conventional histograms of motion vectors by fuzzification to give a more continuous description of angular motion patterns sensitive to small displacements and robust to variations of illumination and motion intensity. The

low computational cost of the method and its emphasis on directional motion enable micro-movements in facial areas to be efficiently analyzed. Comparisons of the CASME, CASME II, and SMIC datasets revealed better accuracy than with conventional methods such as LBP-TOP. The sophisticated motion representation in FHOFO makes it a strong candidate for feature extraction in deepfake detection, especially for identifying subtle direction discrepancies in faked videos.[3]

- Zhou et al. (2021) emphasized learning discriminative and salient motion-based characteristics for recognition of micro expression in their paper "Feature Refinement (FR) Framework for Micro-Expression Recognition."It uses optical flow features between onset andapex frames to highlight motion change and refines expression-specific features with attention driven learning. Considering also expression-specific feature distillation and fusion, FR has been able to outperform the benchmarks, CASME II, SMIC and SAMM. It offers a smaller architecture andattention ones, which deliver computing efficiency while not compromising on high precision of results. The attentiveness of this framework towards critical faces' regions and amplification of motion features indeed confirms its effectiveness in identifying slight facial abnormalities and hence, bridgesthe gap-resided between micro-expression recognition and deepfake detection.

2.8 Hierarchical Transformer Network (HTNet) for Micro-Expression Recognition

The proposed HTNet, a Hierarchical Transformer-based Network for micro-expression recognition exploiting the optical flow images,the facial region segmentation and the hierarchical attention, is shown to outperform existing facial flow based approaches. Micro-expressions aresubtle and transient, making them ideal targets for the action model which focuses on the reconstruction of complex facial dynamics not captured by conventional methods. HTNet Architecture — Grounded in the integration of spatial and temporalinformation for the enhancement of the recognition performance and is particularly fit for micro-expression analysis and deepfake detection tasks.[7]

2.8.1 Optical Flow and Kinematic Analysis

HTNet naturally leverages optical flow, which is amotion analysis technique that calculates temporal and spatial frame differences. HTNet highlightsthe most useful motion information regarding micro-expressions by calculating optical flow maps at the frame level between onset to apex frames. It also utilizes optical strain, a derivative of optical flow, to measure the intensity and direction of small scale motionvariations. These features give the three-dimensionalconfiguration of motion dynamics in horizontal, vertical, and strain orientations. Fig. 7a shows that the D&C-RoIsmethod is able to correctly identify the apex frame of the most expressive micro-expression, with precise temporal synchronization. HTNet (Hybrid Terrestrial Network) operates by segmenting significant facial regions,namely the left eye, right eye, left lip, and right lip, depending on landmarks provided by the MTCNN (this faction involved in the mobility process). HTNet alleviates artifacts from unwanted motions (e.g. background artifacts and headmovements) by segmenting optical flow

maps into specific areas that only focus on local facial movements 14. By performing this localisation, the model is able to investigate minute muscle distortions that are critical to being able to identify micro-expressions.

2.8.2 Hierarchical Attention Mechanisms

HTNet uses hierarchical attention methods based on localised and global attention for feature extraction. Detail-oriented Attention (Second Bottleneck): Gathers fine-grained details within each area of a single facial region, such as slight muscular contractions and expansions. High Level Attention: Use Regional Attention Lines (performed by concatenating p(Ai)+p(Bi)+... $\rightarrow$ p(R0)) to gather interactions suchas moving eyes and mouth coordination to represent the dynamic of the whole face. BlockAggregation: Merges smaller includes blocks of features to form bigger representations that help the model merge local dependencies within a particular context with global linkages between regions. Incorporating this hierarchical attention approach, HTNet can appropriately capture the subtle characteristics ofmicro-expressions and the dependencies across facial regions, which further enhances its ability to learn complex patterns.

2.8.3 Rationale for Employing Micro-Expressions in Deepfake Identification

Micro-expressions are involuntary and subtle facial movements that reveal true emotions, and they can't be faked. Thisis precisely the logical argument that could be used to apply them to the identification of deepfakes. Most of the generated adversarial network (GAN)-based models are charged withlearning the overall characteristics of faces, and lack the sensitivity of micro-movements.

Traditional approaches to deepfake detection are contingent on the appearance of visual artefacts, inconsistencies in time, or pixel anomalies which themselves don't perform too well as the quality of GANs evolves. Micro-expressions add one dimension of analysis that focuses on subtle, movement-based irregularities that aren't easily identifiable by artifact-based schemes.HTNet's dependence on temporal dynamics and spatial attention makes it well-suited to leverage micro-expressions for detecting deepfakes. HTNet captures anomalies in facial dynamics that are not replicable by imitation models by inspecting temporal interactions between facial areas.

Hierarchical Attention and Region-Specific Modelling The model, on a basic level, finds localized features, such as slightcontractions or expansions of facial muscles in each region of interest. It models interactions between facial areas, including the synchronizationof lip and eye movements. Suchhierarchical representations allow HTNet to identify both localised anomalies (e.g. abnormal eye twitching) as well as global inconsistencies (e.g. uncoordinated movements across areas).

2.8.4 Potential Application to Deepfake Detection

Deepfake videos seldom do agood job of reproducing the nuanced dynamics of human facial expressions, especially micro-expressions. HTNet's capability of modeling subtle facial motions in conjunctionwith its region-specific attention mechanisms make it a perfect candidate framework to tackle this limitation. Sharinga similar philosophy of looking for consistency, HTNet provides a compelling and rational approach to determine if the media is synthetic by exploring how facial regions interact with each other and evolve over time through the existing relationships between these regions. By utilizing optical flow and hierarchical modelling it captures thefine grained details that are essential for detection establishing a connection between micro-expression detection and deepfake images. [19]

Chapter 3

Description of model

3.1 Introduction

In this research, we used HTNet for microexpression analysis and then detected deepfake content through transfer learning and without transfer learning based on the HTNet microexpression analysis. HTNet uses a hierarchical framework with multiple levels and processes facial features at various scales. This design allows the model to evaluate precise elements in small areas.
HTNet is designed with transformer layers and a block aggregation process, arranged in a hierarchical structure. Thus, HTNet consists of:

1. Transformer layers
2. Block aggregation process

Each transformer layer applies self-attention to individual image blocks, allowing the model to capture fine-grained features in the lower layers. The block aggregation step then combines smaller blocks into larger ones, enabling interactions between blocks within the same layer and capturing broader, coarse-grained patterns. All blocks within a single layer share the same set of parameters for consistent processing.
Finally, a Multi-Layer Perceptron (MLP) processes the final feature maps to classify micro-expressions. This modular, hierarchical approach allows HTNet to effectively extract and integrate features across different scales and levels of detail, enhancing its performance in micro-expression recognition.

3.2 HTNet Architecture

The HTNet architecture processes input images through a hierarchical workflow composed of three interconnected blocks, each responsible for distinct operations that extract, aggregate, and classify features.
In the initial phase, the Patch Embedding Layer (Block 1) partitions the input image, typically sized $224 \times 224 \times 3$, into small patches of dimension $P \times P$. Each patch undergoes a linear projection to produce an embedding matrix $X \in R^{b \times n \times d}$, where b is the batch size, n represents the number of patches ($H \times W/P^2$), and d is the embedding dimension. To preserve spatial relationships, trainable positional embeddings are added.

Next, in Block 2 (Attention + MLP and Feature Aggregation), multiple Transformer layers process the embeddings. Each Transformer layer includes Layer Normalization (LN), defined as:

$$LN(x) = \frac{x - \mu}{\sigma}\gamma + \beta, \tag{3.1}$$

Multi-Head Self-Attention (MSA), computed as:

$$MSA(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V, \tag{3.2}$$

where Queries (Q), Keys (K), and Values (V) are linear projections of the input embeddings, and a Feed-Forward Network (FFN):

$$FFN(x) = \max(0, xW_1 + b_1)W_2 + b_2. \tag{3.3}$$

Residual connections are integrated after each sub-layer to improve gradient flow and training stability.

In the final stage, Block 3 (Deep Feature Extraction and Global Representation) employs additional Transformer layers to capture deep hierarchical features and global dependencies. Outputs from multiple attention heads are concatenated and projected through a linear layer. Lastly, the aggregated representation $X_{cls} \in R^{b \times d}$ is fed into a Fully Connected Classification Head Layer, which applies a softmax activation function to generate class probabilities.

This hierarchical design, from local patch embeddings to deep global representations, allows HTNet to effectively extract multi-scale features and enhance micro-expression recognition performance.

3.2.1 LayerNorm

This layer normalizes the inputs in a mini-batch.

$$y = \frac{x - E(x)}{\sqrt{Var(x) + \epsilon}} \times \gamma + \beta \tag{3.4}$$

The formula is used to normalize the data by subtracting the mean and dividing it by the standard deviation. This contributes to the stabilization of the training process by ensuring that the inputs to the layers in the network have a mean of zero and a standard deviation of one. The formula contains learnable parameters that can be used to scale and shift the normalized data.

The custom implementation of LayerNorm standardizes the input across the channel dimension. For normalization, the layer calculates the mean and variance of the input features across the channel dimension. To shift the normalized output, it uses learnable parameters gamma (gain) and beta (bias), which help stabilize the learning process.

3.2.2 PreNorm

This layer ensures that the input to subsequent layers does not shift too much. This is achieved by applying another function—in our case, a feed-forward layer—after normalization to ensure stability and convergence. This layer essentially acts as a wrapper module, where LayerNorm is passed to fn, which is a neural network function.

3.2.2.1 First Convolutional Layer

This layer expands the dimensions of the input data. It transforms the input feature map from dim (number of channels in the input feature map) to dim $\times$ mlp_mult. Here, mlp_mult is a multiplier that determines the expansion factor for the middle hidden layer. The expansion allows the network to create a richer representation of the input features, enabling the model to capture more complex patterns and interactions within the data.

3.2.3 FeedForward Layer

This component of the transformer architecture contains multiple layers designed to work within transformer blocks following the self-attention mechanism:

- **First Convolutional Layer:** This layer expands the dimensions of the input data. The parameter mlp_mult determines the expansion factor for the middle hidden layer, allowing the network to create a richer representation of the input features, thereby capturing more complex patterns and interactions.
- **Activation Function:** The Gaussian Error Linear Unit (GELU) activation function is used immediately after the first convolutional layer to introduce nonlinearity.
- **Dropout:** A regularization technique used to prevent overfitting, applied after the activation function.
- **Second Convolutional Layer:** This layer reduces the dimensionality back to its original dimensions, processing each position independently.
- **Second Dropout:** Another dropout layer follows the second convolutional layer to further prevent overfitting.

The primary function of this layer is to further process the output of the self-attention mechanism. This network processes each position independently, enabling more complex feature interactions at each position while applying non-linear transformations to the data.

3.2.4 Attention Layer

This layer implements the multi-head self-attention mechanism, designed to capture dependencies across different parts of the input. The process involves a single convolutional layer that divides its outcomes into three parts, often referred to as Query, Key, and Value (Q, K, V):

- **Query, Key, and Value Projections:** These projections transform the input features into a form suitable for computing attention scores.
- **Scaling by Dimension:** The attention scores are scaled by the dimension of the keys to stabilize the gradients during training. This is mathematically represented as:

$$\text{Scaled attention} = \frac{QK^T}{\sqrt{d_k}} \tag{3.5}$$

where d_k is the dimension of the keys.

- **Softmax and Dropout:** After scaling, a softmax function is applied to the attention scores to obtain a distribution of weights across the input features. A dropout layer is then applied to prevent overfitting by randomly omitting certain elements from the output vector.

3.2.5 Aggregate Layer

The aggregate layer is a utility function that performs several operations to downsample the spatial dimensions of feature maps. This process prepares the data for sequential processing by subsequent layers at a more abstract hierarchical level.

- **Convolution Layer:** This layer preserves or modifies the number of channels, on average, by a factor of growth to capture more complex features.
- **Normalization and Pooling Layer:** Standard normalization is paired with a pooling operation to reduce dimensions.
- **Transformer Layer:** This layer consists of a stack of attention and feed-forward modules. It also adds positional embeddings to retain spatial information.

3.2.6 HTNet Layer

HTNet is the full model architecture where images pass through a hierarchy of transformer blocks:

- **Patch Embedding:** This step transposes the input image into a channel token over larger dimensions, making it suitable for transformer-based processing.
- **Hierarchical Processing:** The model utilizes several transformer blocks at varied levels, followed by multiple aggregation steps to iteratively compress the feature representation.

3.2.7 MLP Head

The final linear classifier in the architecture pools global features to make class predictions.

3.3 Optical Map Extraction

To generate an optical feature map, identifying both the onset and apex frame indices is crucial. Since micro-expression datasets already indicate the onset frame index, only the apex frame index needs to be determined from the video sequences. We

use the D/C-Rolls method, a widely adopted approach in previous micro-expression studies, which maps the relationship between the onset frame and facial moving frames, enabling precise detection of the apex frame. As a result, optical flow features are accurately extracted for micro-expression recognition tasks.
In this formula:

$$d = \frac{\sum_{i=1}^{B} h_{1i} \times h_{2i}}{\sqrt{\sum_{i=1}^{B} h_{1i}^2 \times \sum_{i=2}^{B} h_{2i}^2}} \tag{3.6}$$

BBB represents the number of histogram bins, h_{1i} corresponds to the first frame, and h_{2i} refers to all frames excluding the first. The frame with the greatest difference in ROIs is chosen, indicating the apex frame with maximum facial muscle movement. Next, optical flow feature maps are created from the onset and apex frames. The optical flow feature map can be mathematically expressed as:

$$V(x, y, t) = (u, v) = \left(\frac{dx}{dt}, \frac{dy}{dt}\right), \quad x, y \in \{1, 2, ..., X\}, \quad y \in \{1, 2, ..., Y\} \tag{3.7}$$

X and Y denote the frame's width (W) and height (H), respectively, and the optical flow vector VW is represented as $[V_x, V_y] \leftarrow X, V_y][V_x, V_y]$, with dimensions $RW \times H + 2R(W \times H$ times $2)RW + H + 2$. Our approach calculates variations in the optical flow field, known as optical strain, by using first-order derivatives, which measure the extent of facial displacement, revealing subtle facial movements crucial for micro-expression analysis. By computing optical strain, we capture detailed facial dynamics, enhancing micro-expression recognition accuracy.
Optical strain $V2V\_V2Z$ is calculated using partial first-order derivatives:

$$V_i = \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y}, \quad V_z = \frac{\partial V_x}{\partial y} + \frac{\partial V_y}{\partial x} \tag{3.8}$$

representing variations in horizontal and vertical flow components. The final three-dimensional optical flow feature map ($Vm(V_m)(Vm)$) combines horizontal, vertical, and strain components $[V_x, V_y, V2I/V_z, y, V_x]/V_2I/V2Z$, with dimensions $RWh + 3R * (W$ times H times $3)RW + H + 3$.
Our model uses a region-specific approach, analyzing four facial areas—the left eye, right eye, left lip, and right lip—instead of the full face. This targeted approach reduces interference from background noise, such as flickering lab lights, during recordings.
To extract facial-region optical flow maps from Vm/V_m, we use Multi-task Cascaded Convolutional Networks (MTCNN) to detect facial landmarks in apex frames. From Vm/V_m, we crop four region-specific optical flow maps: left-eye, right-eye, left-lip, and right-lip feature maps. Each cropped feature map is half the size of the full optical flow map, with dimensions $W2 + H2 + 3frac(W2)(2$ times $H2)(H2 + 2)times32W + 2X + 2H3$.
After extracting the four optical flow maps, we merge them and input the combined features into HTNet for micro-expression recognition. A visual representation of the entire process is provided in Figure 1.

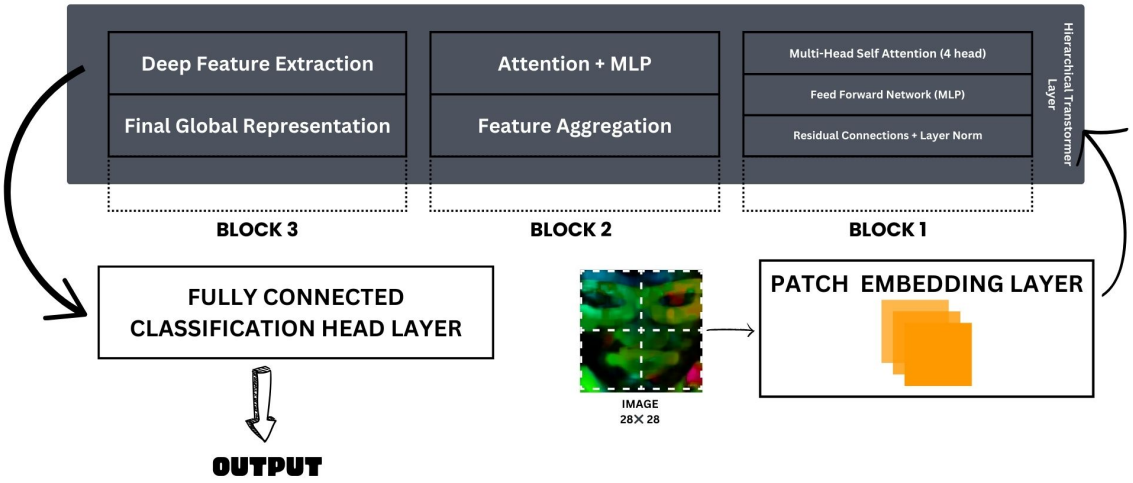

Figure 3.1: Model Architecture

3.4 Transformer Layer:

In this approach, the input optical flow image, sized $H \times W \times 3$, is divided into image blocks with dimensions of $P \times P$. After applying linear projection and partitioning, each patch acquires a feature dimension of $P \times P \times 3$. These patches are then flattened into an input tensor, represented as:

$$X \in R^{b \times H_n \times n \times d} \tag{3.9}$$

where H_n denotes the number of blocks per level, b is the batch size, n indicates the sequence length, and:

$$H_n \times n = \frac{H \times W}{P^2}. \tag{3.10}$$

Within each block, multiple transformer layers are utilized, with the number of layers determined by the hierarchical level. Each transformer layer includes Layer Normalization (LN), a Multi-Head Self-Attention (MSA) mechanism, and a Feed-Forward Fully Connected Network (FFN).
To capture spatial relationships, a trainable positional embedding vector is added to all sequence vectors in R^d, encoding spatial and positional information effectively. The transformer operations are defined as follows:

$$Y_t'^i = Y_t^{i-1} + MSA(LN(Y_t^{i-1})) \tag{3.11}$$
$$Y_t^{i+1} = Y_t'^i + FFN(LN(Y_t'^i)) \tag{3.12}$$
$$Y_t'^{i+1} = Y_t^{i+1} + MSA(LN(Y_t^{i+1})) \tag{3.13}$$
$$Y_t^{i+2} = Y_t'^{i+1} + FFN(LN(Y_t'^{i+1})) \tag{3.14}$$

where l represents the block index within a hierarchy layer, and L is the total number of blocks per hierarchical layer. The FFN comprises two layers, defined as:

$$\max(0, xW_1 + b)W_2 + b. \tag{3.15}$$

In each block, the MSA mechanism processes the input $X \in R^{n \times d}$ by generating queries (Q), keys (K), and values (V), where n is the sequence length, and d is the input dimension. Scaled dot-product attention is computed as:

$$MSA(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V. \tag{3.16}$$

The LN operation normalizes each block using:

$$LN(x) = \frac{x - \mu}{\delta} \odot \lambda + \beta, \tag{3.17}$$

where μ is the feature mean, δ is the feature standard deviation, $\odot$ represents element-wise multiplication, and λ and β are learnable parameters.
After the transformer layer, block aggregation is performed to merge outputs by grouping every four small blocks into a larger block, consolidating feature representations.

3.5 Block Aggregation

The block aggregation mechanism in HTNet parallels several pyramid structures but differs by employing local attention on individual image blocks rather than global attention on the entire image. This local attention is crucial for capturing localized facial muscle movements, essential for micro-expression recognition. HTNet processes optical flow maps within each block independently, with block communication occurring during aggregation.
Specifically, block aggregation fuses local and global features: low-level aggregation exchanges information among four facial regions—left eye, right eye, left lip, and right lip—extracting fine-grained features of facial dynamics. High-level aggregation broadens this exchange, capturing coarse-grained expressions. The optical flow image size transforms from:

$$X_l \in R^{b \times h \times w \times d_l} \quad \text{to} \quad X_{l+1} \in R^{b \times 2h \times 2w \times d_{l+1}}, \tag{3.18}$$

merging all regions into a single facial feature map:

$$X \in R^{b \times H \times W \times D}. \tag{3.19}$$

The block aggregation employs a 3×3 convolutional layer, Layer Normalization (LN), and 3×3 max pooling. Initially, a 3×3 convolution merges partial features into four blocks (2×2 each) for the key facial regions. A subsequent 3×3 convolution exchanges information among these regions, reducing blocks to 1×1 and producing full optical feature maps, which are then passed to an MLP for micro-expression classification.
This hierarchical design captures multiscale facial features, enhancing micro-expression recognition.

Application in deepfake detection

HTNet's ability to capture subtle facial dynamics and regional interactions can be applied to deepfake detection. By analyzing micro-expressions and optical flow patterns, HTNet can identify inconsistencies in facial movements that often appear in deepfake videos. The transformer layers and block aggregation help differentiate between authentic and manipulated facial expressions, making HTNet a valuable tool for video forensics and security applications. To achieve this, we used both transfer learning method and non-transfer learning method.

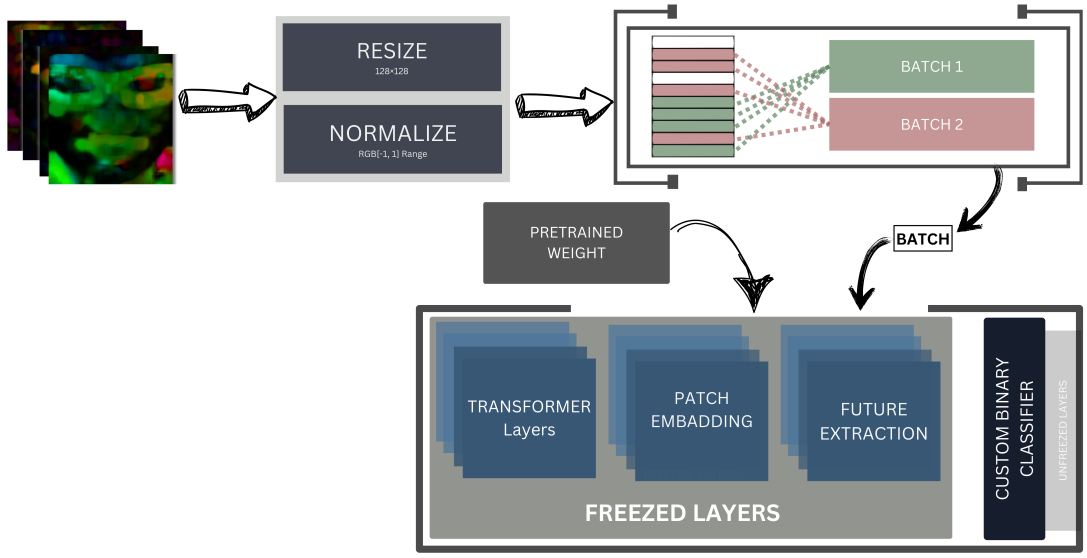

Figure 3.2: Transfer Learning Pipeline

The presented transfer learning architecture is designed for deepfake detection through microexpression analysis. The system utilizes a deepfake dataset for fake samples and a microexpression dataset for real samples, repurposing a model initially trained for microexpression detection to identify deepfakes based on facial micro-movements. First, input images undergo preprocessing: they are resized to 128×128 pixels and normalized to the range $[0, 1]$. The preprocessed images are then divided into batches (e.g., Batch 1, Batch 2), denoted as:

$$X \in R^{b \times h \times w \times c}, \tag{3.20}$$

where b is the batch size, h and w are image dimensions, and c is the number of channels.
Next, the images are passed through a pretrained model with frozen layers, which include Transformer layers, patch embedding, and feature extraction modules. The patch embedding splits each image into $P \times P$ patches and maps them to a D-dimensional embedding space, producing:

$$X_{\text{emb}} \in R^{b \times n \times d}, \tag{3.21}$$

where n is the number of patches. Transformer layers apply multi-head self-attention (MSA) defined as:

$$MSA(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V, \tag{3.22}$$

capturing spatial dependencies. Residual connections and layer normalization further stabilize learning.

The output from the frozen feature extractor is input to a custom binary classifier. The classifier, a fully connected neural network, maps feature embeddings:

$$X_{\text{feat}} \in R^{b \times d} \tag{3.23}$$

to class probabilities using a softmax activation. The binary cross-entropy loss is defined as:

$$L = -\left[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\right], \tag{3.24}$$

which measures classification accuracy.

This transfer learning approach leverages knowledge from microexpression detection to identify subtle facial anomalies indicative of deepfakes, enabling precise and efficient deepfake detection.

Chapter 4

Dataset

4.1 Data Description

Considering that we are discovering deepfake using microexpressions, we usedboth the microexpressions data-set and the deepfake data-set — data that is useful for our goal. We employ the existing datasetand the deepfake dataset for the microexpression dataset. Weuse Faceforensic++ dataset to collect deepfake.

4.1.1 FaceForensics++

Thisdataset contains 1000 original video clips, altered using computerized face editing techniques, Deepfakes, Face2Face, FaceSwap, and NeuralTextures. All of these videos display a mostly frontalface that is not occluded and is fully trackable. It means thatautomated editing tools can produce apparent fakes. [22]
We consider neural texturedataset from Faceforensic++ for our purpose of both fake detection. Neural Textures are made by neural networks that capture complex visual features of human faces such asskin texture, varying light sources, and facial expressions. These images are trained using a real face video set, which can later modifythe faces in other videos, without any detectable modifications.
Established microexpression datasetreflects our list of micro expression recognition dataset. Several important datasets areincluded in the evaluation of microexpression recognition for this Hierarchical Transformer Network paper [5]. Werandomly chose three datasets from our previous work, SAMM, CASME II, and CASME III.

4.1.2 Microexpression Dataset

- **SAMM:** 133 micro-expressions and 147 long videos from28 conditional participants with frame-by-frame Action Units coding. Image crop to 28x28 pixel at200 FPS
- **CASME II:** It contains 145 samples of 24 subjects with a frame rate of 200and a resolution of 640x480 pixels, which are cropped to 28x28 for the experiments.
- **SMIC:** 164 samples from 16 subjects, videos weretaken per frame with 100 fps and resized to 28x28 pixels from 640x480 pixels.

4.2 Data Preprocessing

A traditional method in the field of neural rendering is to utilize a collection of 26 longvideos that include neural texture derived from the FaceForensic++ deepfake dataset. The videos are broken up into different episodes,with each episode labelled based on the different microexpressions seen in the clip. Then, the OpenFace toolkit — a very advanced baseline for facial behaviour analysis — is used to parse theseepisodes and split each into its individual image frames. The converted video allowsfor extensive analysis of microexpressions by slicing the video into static images. This enables you to analyze facial expressions andsubtle emotional cues frame by frame.

4.2.1 Facial Action Coding System and Action Units

Then we extract the facial feature using OpenFace tool which did Facial Landmark DetectionHead Pose Estimation, Eye Gaze Tracking, Facial Action Unit Recognition, Feature Extraction. Facial Action Unit Recognition (FAU) is a fundamental aspect of affective computing and facial expression analysis. It is based on recognising and decoding movements from the identifiable musculature groups in the face — known action units (AUs). These AUs are included in the Facial Action Coding System (FACS), a systematic tool created by psychologists Paul Ekman and Wallace V. Friesen in the 1970s. FACS is widely used in behavioral research to characterize facial movement that is visually distinguishable[23]. Facial expression based emotion detection-Affect is inferred from facial action units (AUs). These AUs can then be utilized with technologies such as OpenFace which connects discrete muscle movements with directly identifiable emotional states. This approach involves collecting facial data, extracting features to identify Action Units (AUs), and subsequently employing either rule-based systems or ML models to associate (AUs) with emotions (for example, happy or angry) [9].

Emotion	**Action Units**
Hapiness	6+12
Sadness	1+4+15
Surprise	1+2+5B+26
Fear	1+2+4+5+7+20+26
Anger	4+5+7+23
Contempt	R12A+R14A

Table 4.1: Action Unit System

These action units are generated on facial landmarks on the basis of FACS. Each detected facial movement is coded by assigning it to one or more AUs. These AUs are further scored based on their intensity, which typically ranges on a scale from 0 (no activity) to 5 (maximum activity). The intensity scoring is crucial as it reflects the strength of the muscle contraction and hence the expressiveness of the emotion. The recorded AU data can then be analyzed to interpret the subject's emotional state, detect subtle emotional responses, or study communication patterns. For instance, certain combinations of AUs might indicate specific emotions like happiness (AU6:

Cheek Raiser and AU12: Lip Corner Puller) or anger (AU4: Brow Lowerer and AU23: Lip Tightener). A combination of AUs is used to recognize emotions. For instance, a smile (AU6 + AU12) might indicate happiness, while furrowed brows (AU4) and a lip press (AU24) might be associated with anger.
consider the detection and analysis of AU12 (Lip Corner Puller), which is associated with smiling.

4.2.2 Onset,Apex,Offset

Detecting the Onset Frame, Apex Frame, and Offset frame is very important forusing deepfake videos to predict microexpression [21]. The Onset Frame is the initial detection of facial musclemovement, the Apex Frame indicates the highest point of expression intensity, and the Offset Frame indicates the transition back to a neutral expression. However, the most important reason to find OnsetFrame,ApexFrame and OffsetFrame is to find the optical flow. It is the pattern of evident motion of objects in the visual field based on the movement ofa camera (or observer) in a scene. It is a 2D vector field Here each vector is the displacement vector which shows how points on the original frameshift to the next frame.

take, for example, the detection and analysis ofAU12 (Lip Corner Puller), which is correlated with smiling: An algorithm calculates a change of distance between two points to the right or left of the lips. [It] is assumed that these points are known to two reference vectors, and we label the positions of the points inthe neutral state p1 and p2, and in an expressed state, p1 prime and p2 prime.

- **Calculation:**The mathematical expression for the movement might be $\Delta d = \|p_1' - p_2'\| - \|p_1 - p_2\|, where\| \cdot \|$ denotes the Euclidean distance between the point.
- **Thresholding:** If $\Delta d\Delta d$ exceeds a predefined threshold, it indicates significant movement characteristic of AU12 activation.

4.2.2.1 Scaling

Size of AU12 activity can be scored on ascale (e.g., 0 to 5), indicating the range of muscle activity detected. This can also be interpreted mathematically as Intensity=$[k\cdot \Delta d$Intensity=$[k\cdot\Delta d]$, where the variable k is a scaling factor that defines the distance between locations in terms of intensity score

4.2.2.2 Integration with other AUs

AU12 Will Not Happens Alone: AU12 is not theone and only AU. AU12 also has its own vocabulary, in the same way its companion AUs do, such thatthe presence of AU12 or the intensity of AU12 (e.g. max intensity AU12 as AU12) with other AUs like AU6 (Cheek Raiser) can signify different expressions (i.e. genuine smile vs polite smile).

4.2.3 Optical Flow

Optical flow is the observed motion of image objects in a visualscene caused by the motion of the object or the observer, between two consecutive frames of a sequence. It is a2D vector field. Here each vector is a displacement vector, which describes the way a point moves from the first frame to the second frame. One of the assumptions of opticalflow is that the pixel intensities of an object do not change from frame to frame. Next, similar to neighbouringpixels, the mobility of neighbouring pixels.

Consider a pixel $I(x, y, t)$ in the first frame (notice the addition of a new dimension, time, t). Earlier, we were working with images only, so time was not considered. Now, if the pixel moves by a distance (dx, dy) in the next frame taken after dt time, we assume the pixel intensity remains unchanged, leading to the equation:

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \tag{4.0}$$

Taking the Taylor series approximation of the right-hand side, removing common terms, and dividing by dt, we obtain the following equation:

$$f_x u + f_y v + f_t = 0 \tag{4.0.1}$$

where:

$$f_x = \frac{\partial f}{\partial x}, \quad f_y = \frac{\partial f}{\partial y}$$

$$u = \frac{dx}{dt}, \quad v = \frac{dy}{dt}$$

Above equation is called the Optical Flow Equation. In this equation, f_x and f_y represent image gradients along the x and y directions, respectively. Similarly, f_t is the gradient along time. However, (u, v) is unknown. Since we have a single equation with two unknowns, this system is **underdetermined**. Several methods have been proposed to solve this problem, one of which is the **Lucas-Kanade method**.

There exists various ways to create optical flow. Lucas Kanade method, denseflow, Gunner farnbeck algorithm We specially chose gunner farnbeck algorithm since it is a dense optical flow estimation method that estimates chance fields among consecutive frames and It is optimized and appropriatefor actual time uses.

Comparison with Other Optical Flow Methods

Farnbeck Optical flow is best suited for microexpression videos, as we are using videos which are converted into image frames later andthe speed is fast and low the complexity. However for deepfake videos it more complex and complex so we use RAFT.

4.2.4 Gunner-Farnebeck for Micro-expression

One example of a dense optical flow method is the Farneback algorithm. This means that, unlike other methods that focus on specific features such as corners,

Method	Accuracy	Speed	Complexity	Best Use Case
Farnebäck Optical Flow	Low-Medium	Fast	Low	Real-time, simple motion
Lucas-Kanade	Low-Medium	Fast	Low	Sparse feature tracking
DeepFlow	High	Medium	High	Dense motion with accuracy
TVL1	Very High	Slow	Very High	Robust deep learning-based flow
RAFT	Very High	Slow	Very High	Best accuracy for complex scenes

Table 4.2: Comparison of Optical Flow Methods based on Accuracy, Speed, and Complexity

it calculates the motion for every single pixel in an image. It uses a list of flow vectors$(dx/dt, dy/dt)$ to determine the optical flow's magnitude and direction. Then displays the flow's angle (direction) by hue and magnitude (distance) via HSV colourrepresentation value. The HSV value is set to themaximum of 255 for best visibility. OpenCV has a function called cv2 which isused to investigate this dense optical flow. calcopticalFlowFarneback.

Figure 4.1: Optical flow generation

4.2.5 RAFT

Unlike RAFT, which performed correlation by first mappingthe reference and pixel features across space, making the whole process more structured and iterative, and allowing it to better capture small facial deformations and inconsistencies. This is important forspotting deepfakes; subtle differences in facial muscle movement and expression change can be an indication of synthetic manipulation. Furthermore, RAFT is more robust to variations in quadrant lighting, texture, and compression artifacts that are often found in deepfake videos. This makes it a more reliable

approach for motion-based feature extraction. The detection model can employ RAFT to quantify the differences in facial movement from frame to frame, and spot both abnormal motion patterns that deepfake generators have difficulty copying, as well as how facial features move in 3D space across frames.

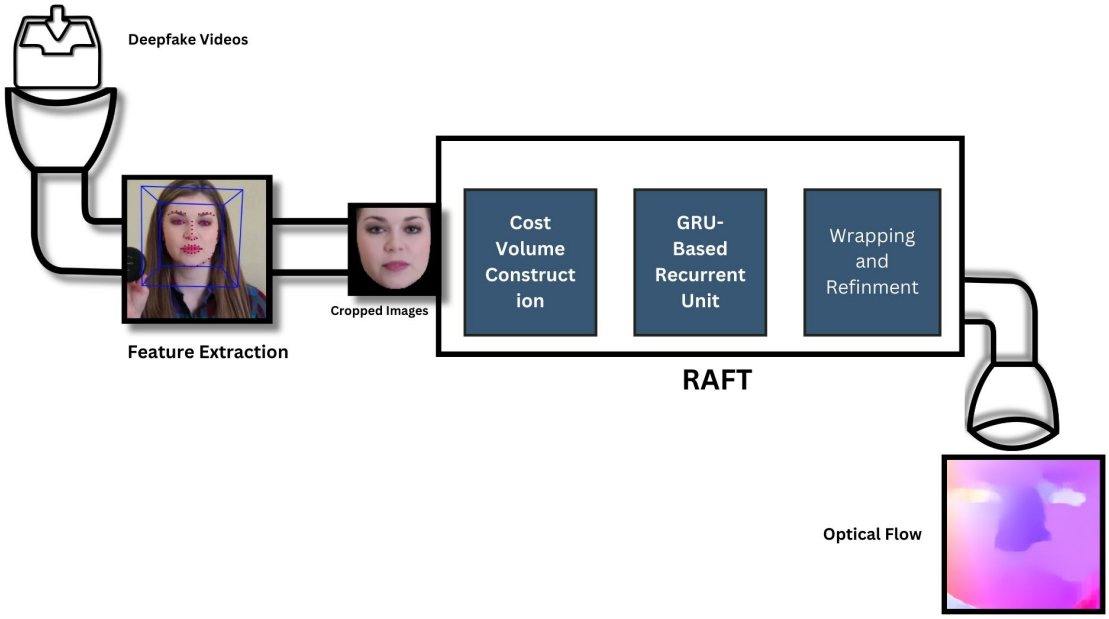

Figure 4.2: RAFT optical flow

4.3 TVL1 Optical Flow

The core of the TV-L1 optical flow method is based on minimizing an energy functional that balances data fidelity and regularization. It employs an L_1 norm for the data term, ensuring robustness against outliers, and total variation (TV) for regularization, which preserves motion discontinuities. The energy functional is given by:

$$E(u) = \int_{\Omega} |\nabla u_1| + |\nabla u_2| + \lambda |I_1(x+u) - I_0(x)| \tag{4.1}$$

where $u = (u_1, u_2)$ represents the optical flow field, and λ controls the trade-off between smoothness and data fidelity. To solve this, the problem is reformulated using a convex relaxation:

$$E_\theta(u, v) = \int_{\Omega} |\nabla u_1| + |\nabla u_2| + \frac{1}{2\theta}|u - v|^2 + \lambda |I_1(x+v) - I_0(x)| \tag{4.2}$$

where v is an auxiliary variable that enables an alternating minimization scheme. The minimization process consists of two steps:

1. **Total Variation Minimization:** With v fixed, the energy is minimized with respect to u using a dual formulation inspired by the Rudin-Osher-Fatemi model.

2. **Data Fidelity Update:** With u fixed, v is updated by thresholding the optical flow constraint.

For our findings we use multiple optical flow generation methods to find the best suited result. We preprocessed both real and fake dataset and generated their optical using all these three methods. The best result is then analyzed.

4.4 Data Analysis

Optical flow captures the movement of pixels between consequent frames, aiding in the identification of natural facial movements vs manipulated ones. The flow distribution of both real andfake videos are analyzed to see whether the deepfake models will produce smooth and natural motion which is consistent with the human body movement pattern.

By contrast, in real videos, the optical flow vectors adhere to natural muscle contractions and exhibit continuous and smooth motion.

That would be sequential on digital basis but judiciously dynamic on human nuances, hence, in the frame of deepfake videos, the gestures seem abruptly changed or over-smoothened without the details of micro-expressions.

As deepfake videos usually lose micro-expression or have artificial jerky motion in optical flow map, therefore it can be a way of detection.

In real micro-expressions, intensity rises gradually, groups of muscles move in synchrony.

Deepfake videos frequently exhibit irregular strainpatterns, with facial muscles appearing either static or exaggerated in a way that seems unnatural.

4.4.1 Mean Optical Flow Magnitude (MOFM)

Mean Optical Flow Magnitude (MOFM) is a measurement of the averagedisplacement of pixels. Similarly, MOFM values of deepfake videos are typically smaller, which means facial motion isstiff and unreal.

4.4.2 Optical Flow Variance

It inspectsthe degree of pixel movement change in time. For natural emotions, the variance in real micro-expressions is high and dynamic but with those fake facesmore variance will lead to a smooth transition or lower variance in micro-expressions as they accurately when in the lower.

4.4.3 Facial Action Unit (FAU) Correlation

By extracting FAUs using OpenFace you can compare the actual muscle movements involved between real and deepfakevideos. Deepfake faces have incorrect FAU cor-

relations, which makes them incapable ofreplicating real expressions.

Deepfakevideo seems unnatural due to pathological optical flow which contradicts reality and a lack of micro-expression. This is what makes optical flow-based micro-expression analysis a robust approach foridentifying deepfakes. The hierarchical attention model used by HTNetmakes it easy to detect differences in facial muscle activity, paving the way to distinguish between deepfakes and real movies. Future work should consider multi-modal methods, fusing lip-sync and audio basedanalysis to boost detection performance.

Chapter 5

Methodology

5.1 Methodology

5.1.1 Dataset Preparation and Preprocessing

5.1.1.1 Selection of Datasets and Optical Flow Generation

Micro-Expression Datasets:

- The generated optical flows are utilized to train the HTNet model for microexpression recognition.
- Farneback Optical Flow is utilized for micro-expression image frames due to its efficiency in capturing small, rapid motions.
- Datasets:
  - CASME II (Chinese Academy of Sciences Micro-Expression II): Approximately 146 optical flows.
  - SAMM (Spontaneous Micro-Expression Database): 163 optical flows.
  - SMIC (Spontaneous Micro-Expression Corpus): 132 optical flows.
- The HTNet model uses these small datasets to run over 800 epochs for pretraining in microexpression recognition. For deepfake detection, in case of real dataset we use CASME II dataset containing 13685 optical flows determined from cropped images as real dataset for the purpose of training and testing. Training 80% and testing 20%..

Deepfake Dataset:

- RAFT (Recurrent All-Pairs Field Transforms) and TVL1 is used for deepfake videos to achieve high-accuracy motion estimation across frames.
- FaceForensics++: Approximately 13685 optical flows for neural textures image frames trated as fake dataset for the purpose of training and testing. Training 80% and testing 20%. The dataset is subsequently split equally into fake and real datasets for transfer learning.

5.1.1.2 Hierarchical Region Extraction

- **Local Feature Extraction:** The face is segmented into distinct regions of interest (eyes, lips, and nose) for both microexpression and deepfake images.
- **Global Feature Extraction:** The entire face is analyzed for discrepancies between regions in both microexpression and deepfake images.

5.1.1.3 Normalization & Augmentation

- Resizing frames to 28×28 pixels in alignment with HTNet requirements.
- Data Augmentation: Applies only to the deepfake dataset, including random Gaussian blur, contrast adjustment, flipping, and rotation to enhance robustness.

5.1.2 Preprocessing

- **Optical Flow Calculation:** Optical flow maps are generated for all frames to capture the dynamic facial movements.
  - Farneback algorithm for micro-expression datasets due to its efficiency in detecting small, fast facial movements.
  - RAFT, TVL1 algorithm for the deepfake dataset for both original sequence(real) and neural texture(fake) to accurately capture detailed motions, especially in manipulated videos.
- **Labeling:** Frames are labeled based on the presence of micro-expressions or indications of video manipulation.
- **Data Augmentation:** Includes random rotation, scaling, and horizontal flipping to enhance model robustness against variations in video quality and presentation.

5.1.3 Deepfake Detection via HTNet Adaptation

- HTNet, originally designed for microexpression recognition by capturing subtle facial muscle movements, is adapted to detect deepfakes. It is effective at identifying temporal anomalies in microexpressions, which deepfake algorithms often fail to reproduce accurately.
- The adaptation involves maintaining the learned microexpression characteristics while tuning the model to differentiate between authentic and fabricated films.

5.1.3.1 Freezing Pretrained Layers

- The base layers of HTNet, already trained on SAMM and SMIC datasets, capture essential facial muscle movements and are kept frozen to prevent retraining the entire model.
- Frozen Layers include:

  - Patch Embedding Layer (converts images into patches for processing).
  - Hierarchical Transformer Blocks (apply local and global attention mechanisms).
  - Self-Attention and FeedForward Layers (sensitive to spatiotemporal changes in facial expressions).
  - Optical Flow Feature Extractor (detects motion discrepancies).
- These layers retain critical information on facial dynamics, reducing computational costs and enhancing model generalization.

5.1.3.2 Changing the Classification Head

- The original multi-class classification head of HTNet, designed to classify various microexpressions, is replaced with a new binary classification head for deepfake detection (real vs. fake).
- Old Classification Head:
  - Multiple output neurons for different microexpressions.
  - Utilized softmax activation for classifying multiple emotional states.
- New Classification Head:
  - A single output neuron for binary classification.
  - Uses sigmoid activation to map predictions to a probability scale (0 to 1).
  - Retains layer normalization for stable feature scaling.
  - Includes a global average pooling layer to consolidate high-dimensional features into a single vector for classification.

5.1.3.3 Unfreezing and Fine-tuning the Classifier Head

- Following the replacement of the classification layer, the model undergoes fine-tuning, focusing exclusively on the new classification head while keeping the pretrained layers frozen.
- This approach leverages the hierarchical transformer blocks that excel at capturing informative microexpression features, allowing the model to adapt to deepfake-specific traits without losing its microexpression recognition capabilities.
- Unfrozen Layers:
  - Dense layer (final output layer responsible for binary classification).

5.1.3.4 Adjusting the Loss Function

- The loss function shifts from Cross-Entropy Loss (used for multi-class classification) to Binary Cross-Entropy with Logits Loss, optimizing the model to distinguish between real and fake videos.

5.1.3.5 Optimization Strategy

- The final model configuration involves training only the last classification head with a higher learning rate, while the frozen layers maintain a lower learning rate to preserve the pre-trained features.
- The Adam optimizer is selected for its effectiveness in managing sparse gradients and dynamically adjusting learning rates, thus facilitating efficient learning of deepfake-specific features while retaining microexpression knowledge.

5.1.3.6 Strategy for Training and Fine-Tuning

- The model is fine-tuned on a balanced dataset of real and fake videos from the FaceForensics++ dataset to avoid biases and enhance prediction accuracy.
- The microexpression datasets (CASME II, SMIC, SAMM) continue to be utilized through transfer learning to further refine the model's capabilities.
- The fine-tuning process involves several epochs, with hyperparameters continuously optimized for accuracy, precision, recall, and F-score.

Approach	**Non-transfer learning**	**Transfer Learning**
Frozen Layers	No layers frozen	Patch embedding, transformer blocks, optical flow extractor
Unfrozen Layers	Entire model trained	Only the final classification head is trained
Output Layer	Binary classification (real/fake)	Binary classification (real/fake)
Activation Function	Softmax	Sigmoid
Loss Function	Cross-Entropy Loss	Binary Cross-Entropy Loss
Optimizer	Adam (standard learning rate)	Adam (higher learning rate for classification head)
Training Approach	Train entire model from scratch	Fine-tune only the final classification head

Table 5.1: Comparison of Model Components for Deepfake Detection

5.2 Training Strategy

5.2.1 Transfer Learning/From-scratch learning

Both Pre-trained and not pre-trained models on micro-expression recognition are adapted for the task of deepfake detection. This approach leverages learned spatiotemporal representations to improve detection accuracy with reduced computational overhead.

5.2.2 Fine-Tuning

The model is fine-tuned on a mixed dataset of real and deepfake images to adapt to the specific characteristics of deepfake manipulations.

5.3 Model Training and Evaluation

5.3.1 Loss Functions and Optimizers

- A combination of cross-entropy loss for classification tasks and a custom loss function is used to enhance the detection of subtle micro-expressions and manipulations.
- Optimizers such as Adam are employed for efficient backpropagation during training.

5.3.2 Validation and Testing

- The model is validated using a split of the training data and tested on a separate set of videos from FaceForensics++ to evaluate its performance across different types of deepfakes.
- Performance metrics like accuracy, precision, recall, F1-score, and ROC-AUC are calculated to assess the effectiveness of the model.

Model Training

1. Exhaustive Data Preparation and Preprocessing

Extracting Frames from High-Resolution Frames

For micro-expression recognition, videos from datasets such as CASME II, SAMM, and SMIC are parsed into high-resolution images to retain important facial details.

Normalizations of Higher Levels

Normalizes each frame to ensure that the pixel values are mapped consistently, accommodating variations in lighting conditions or differences in camera sensors.

Normalization Equation

The normalization of image pixels can be expressed with the following equation:

$$X_{\text{norm}} = \frac{X - \mu}{\sigma} \tag{5.1}$$

Equation Explanation

- X - Original image pixel values.
- μ - Mean pixel intensity across the dataset.
- σ - Standard deviation of pixel values.

Purpose

Normalization ensures that the model receives inputs on a standardized scale, preventing overemphasis on specific intensity values.

Contrast Adjustments and Histogram Equalization: These techniques further enhance visibility by adjusting pixel intensity distributions, ensuring that subtle expressions remain discernible.

CNN Feature Extraction Formula

The feature extraction in a Convolutional Neural Network (CNN) can be represented by the following formula:

$$X_{\text{cnn}} = f_{\text{ReLU}}\left(f_{\text{BatchNorm}}\left(f_{\text{conv}}\left(X_{\text{norm}}\right)\right)\right) \tag{5.2}$$

Equation Explanation

- f_{conv} – Applies convolutional filters to extract spatial features.
- $f_{\text{BatchNorm}}$ – Normalizes feature maps to stabilize and accelerate learning.
- f_{ReLU} – Introduces non-linearity, enabling the model to learn complex patterns.

Purpose

This multi-step process enhances facial feature extraction, making the model more robust to lighting variations and occlusions.

Dropout Layers

Dropout is a technique applied during the training of the model to randomly disable neurons, effectively preventing overfitting and ensuring the model's generalization ability. This randomness helps the model avoid learning noise and specific details that might not generalize well on unseen data.

Refined Optical Flow Calculation

Refined optical flow techniques are utilized to capture the temporal dynamics of sequences by computing pixel movements across consecutive frames. This method is particularly useful in recognizing patterns that evolve over time, making it invaluable for applications such as video processing and dynamic scene analysis.

Optical Flow Equation

The optical flow equation used to capture pixel displacement between consecutive frames in a video sequence can be formulated as:

$$V = \text{OpticalFlow}(X_t, X_{t+1}) \tag{5.3}$$

Equation Explanation

- X_t, X_{t+1} – Consecutive frames in a video sequence.
- V – Optical flow vector field representing pixel displacement.

Purpose

Capturing motion vectors allows the model to analyze temporal changes, helping detect micro-expressions based on minute facial movements. This process is essential for applications where understanding dynamics and subtle changes over time is crucial.

5.3.3 Transformer-Based Deep Feature Integration

Multi-Head Self-Attention Mechanism: CNN-extracted spatial features are processed using transformers to focus on the most relevant facial regions dynamically.

Self-Attention Formula

The self-attention mechanism in neural networks is represented by the following equation:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5.4}$$

Equation Explanation

- Q **(Query)** $\rightarrow$ Represents the feature vector being analyzed.
- K **(Key)** $\rightarrow$ Represents different feature vectors from the same image.
- V **(Value)** $\rightarrow$ Stores original feature vector information.
- $d_k \rightarrow$ Scaling factor for numerical stability.

Purpose

The attention mechanism assigns varying weights to different features, ensuring that the model focuses on **critical facial regions** associated with micro-expressions.

5.4 Robust Classification and Prediction

Dense Layers and Softmax Activation: The output of the transformer module is fed into dense layers, culminating in a softmax function to predict the micro-expression class. Softmax Classification Equation:

Softmax Classification Equation

The softmax function is used to compute the probability distribution across different micro-expression classes. It is represented by the following equation:

$$Y = \text{softmax}(W_f Z_{\text{out}} + b_f) \tag{5.5}$$

Equation Explanation

- $Z_{\textbf{out}} \rightarrow$ Final transformed feature vector from the transformer.
- $W_f, b_f \rightarrow$ Weights and biases of the fully connected classification layer.
- $Y \rightarrow$ Probability distribution across micro-expression classes.

Purpose

Converts feature representations into **class probabilities**, enabling accurate expression classification.

Transfer Learning

Selective Parameter Inheritance

- The CNN layers are initialized with weights from a pre-trained model—trained on general face recognition tasks—so that HTNet can benefit from representations already learned.
- A few layers could be reinitialized to allow them to learn the specifics of micro-expression detection.

2. Fine-Tuning with Gradient Monitoring

Adaptive Learning Rate Adjustment

The model dynamically adjusts the learning rate based on gradient behavior to ensure stable convergence and prevent overfitting.

3. Cross-Entropy Loss with L2 Regularization

The loss function incorporating cross-entropy and L2 regularization is given by:

$$L = -\sum_{i=1}^{C} Y_{\text{true},i} \log(Y_i) + \lambda ||\theta||^2 \tag{5.6}$$

Equation Explanation

- $C \rightarrow$ Number of micro-expression classes.
- $Y_{\mathbf{true},i} \rightarrow$ Actual label (one-hot encoded).
- $Y_i \rightarrow$ Predicted probability for class i.
- $\lambda \rightarrow$ Regularization strength.
- $||\theta||^2 \rightarrow$ Sum of squared model parameters.

Purpose

The first term measures prediction accuracy, while the second prevents overfitting by discouraging excessively large weights.

4. Comprehensive Evaluation and Iterative Refinement

Continuous Performance Evaluation

- The model undergoes rigorous validation, where its performance is continuously monitored on unseen data.
- If overfitting is detected, early stopping and additional regularization techniques (e.g., dropout) may be applied.
- Feedback from these evaluations is used to iteratively refine model parameters.

Chapter 6

Result

The best result was achieved using TVL1 optical flow for both real and fake datasets. We used the original sequence from FaceForensics++ for the real dataset and the neural texture from FaceForensics++ for the fake dataset. The HTNet model was run without the pretrained weights for micro-expression recognition. In the comparative analysis section, we will discuss the reasoning behind this choice and demonstrate how using pretrained weights produced different results.

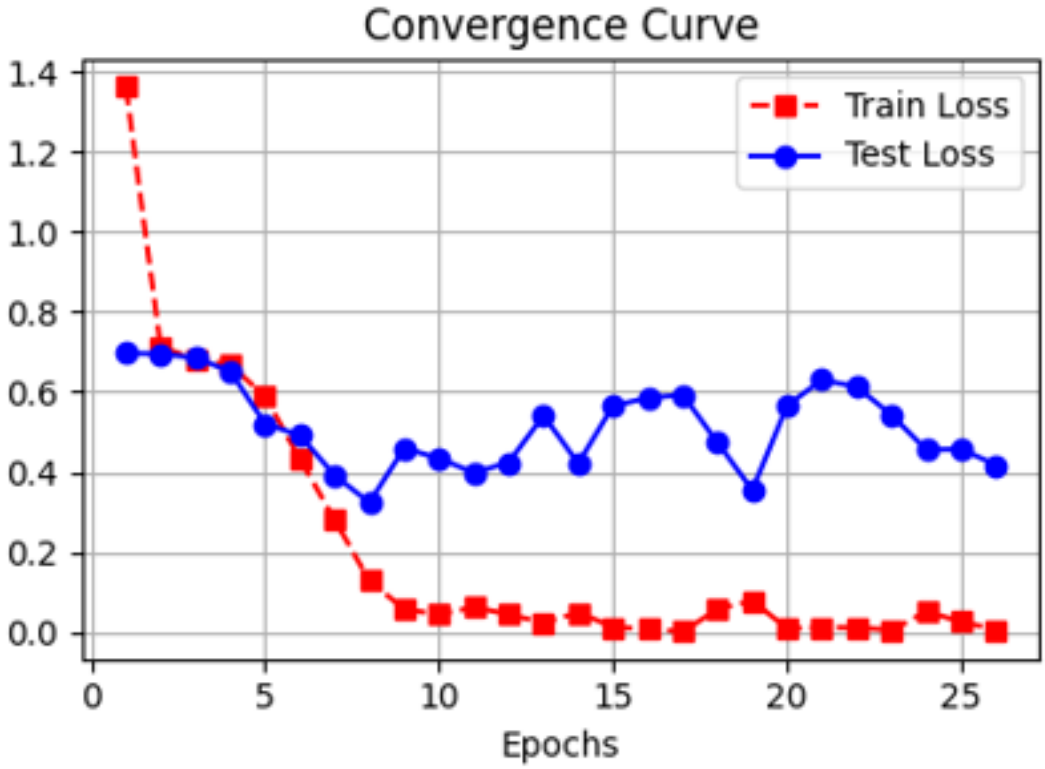

Figure 6.1: Convergence curve of HTNet during training.

The convergence curve shows a steady decrease in training and validation loss. It means that the model is learning effectively. The validation loss follows a similar trend as the training loss.

Performance Metrics of HTNet

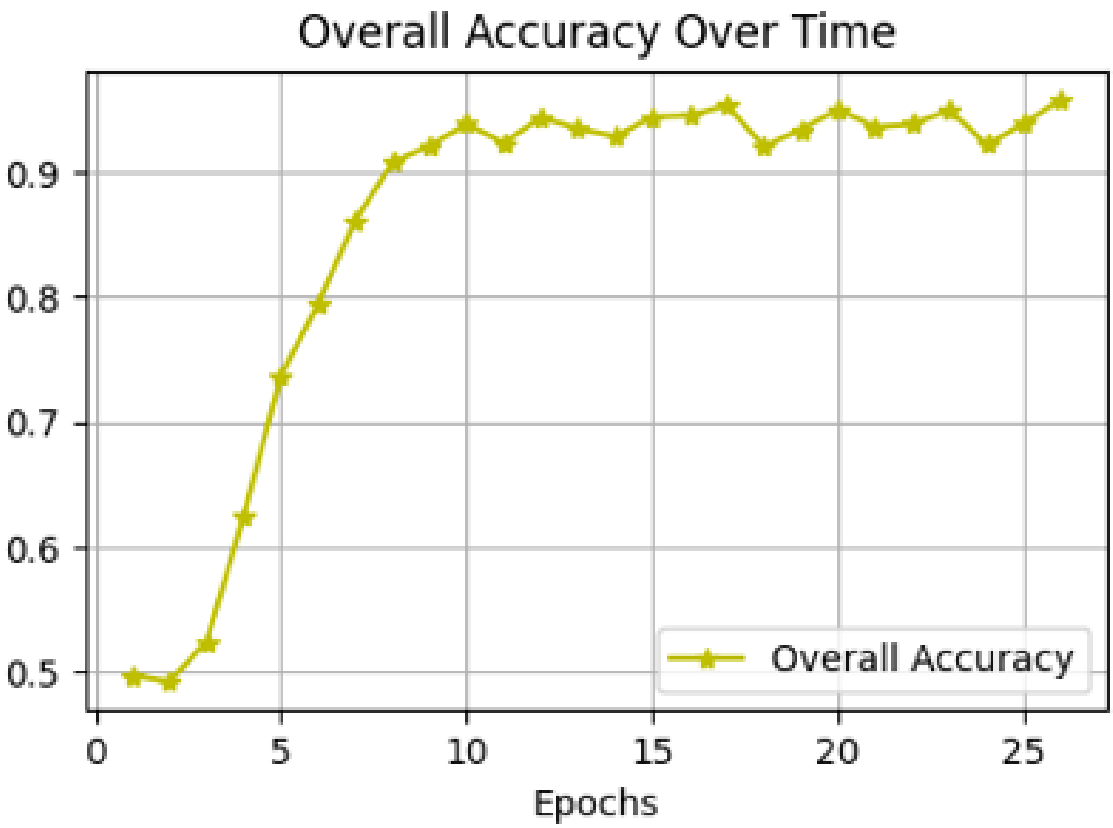

Figure 6.2: Accuracy Curve of HTNet

The accuracy stabilizes around 91-92%. Itshows that the model can differentiate between real and deepfake with relatively high accuracy. The fluctuations are expected due to the complexity of the dataset.

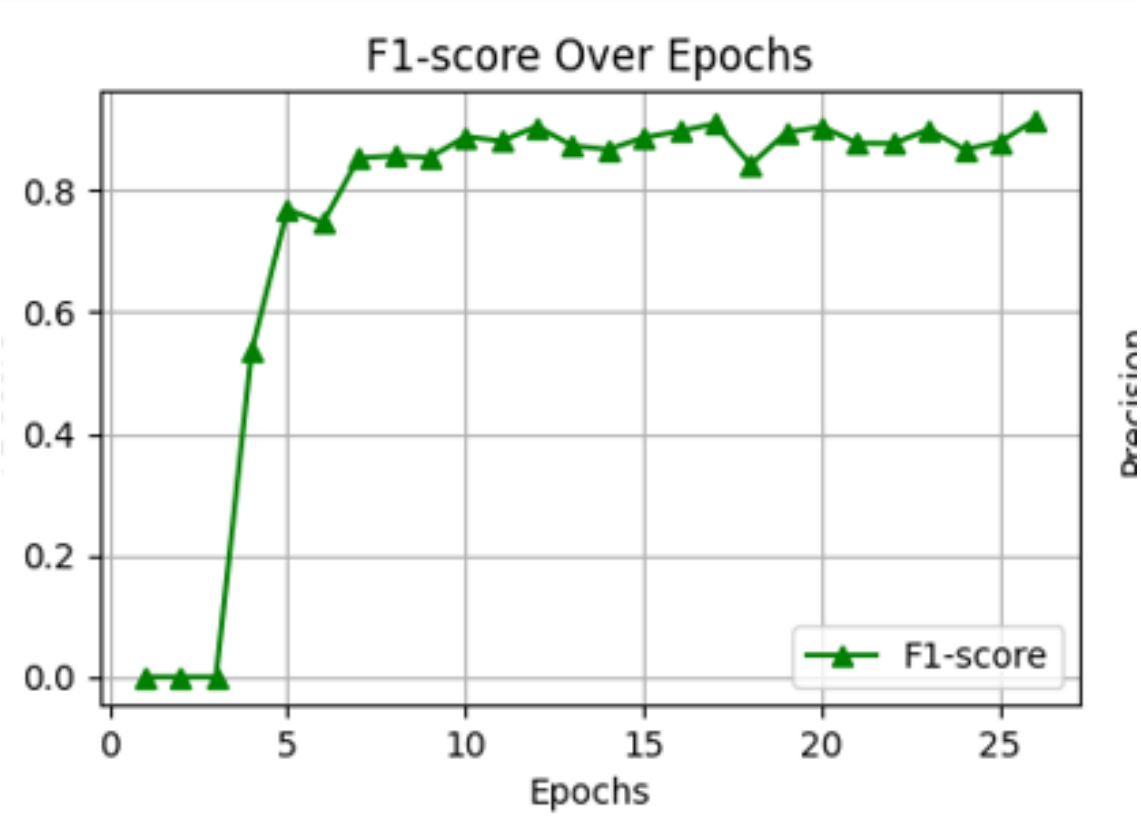

Figure 6.3: F1 Score Curve of HTNet

The F1 Score stays between 0.8 and 0.91. It Balances precision and recall. A stable F1 score suggests that the model maintains a good tradeoff between identifying deepfakes and minimizing false positives.

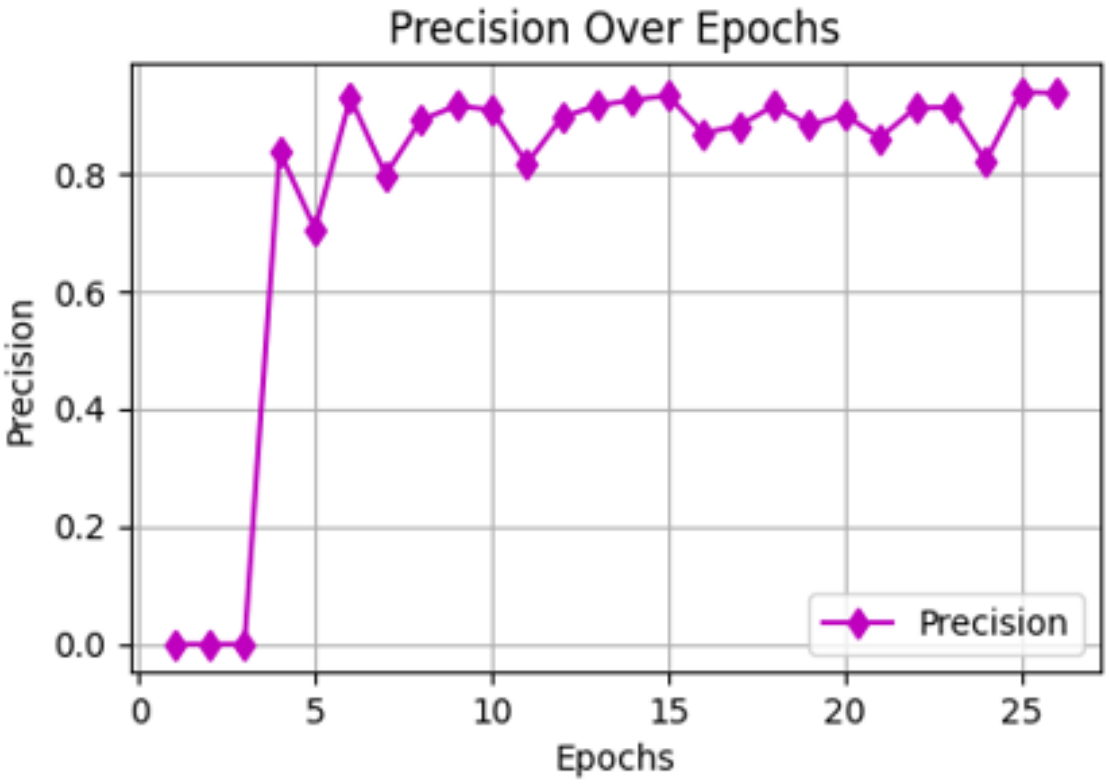

Figure 6.4: Precision Curve of HTNet

Precision remains within the range of 0.8 to 0.91 . It indicates when the model indicates a deepfake, its correct most of the time. Higher precision means fewer false positives, which is beneficial for reducing incorrect deepfake detections.

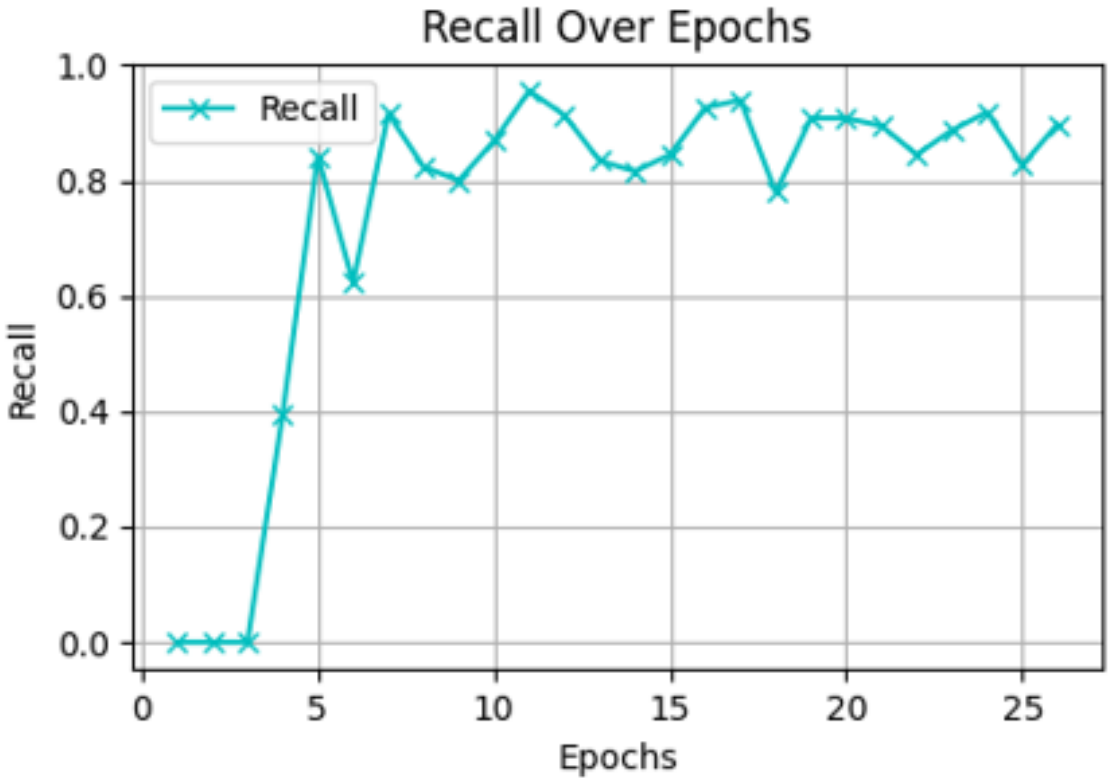

Figure 6.5: Recall Curve of HTNet

The recall score fluctuates between 0.8 and 0.9 . It means the model is capable of detecting deepfakes but occasionally misses some. The interpretation suggests that more training data or fine-tuning could further improve recall.

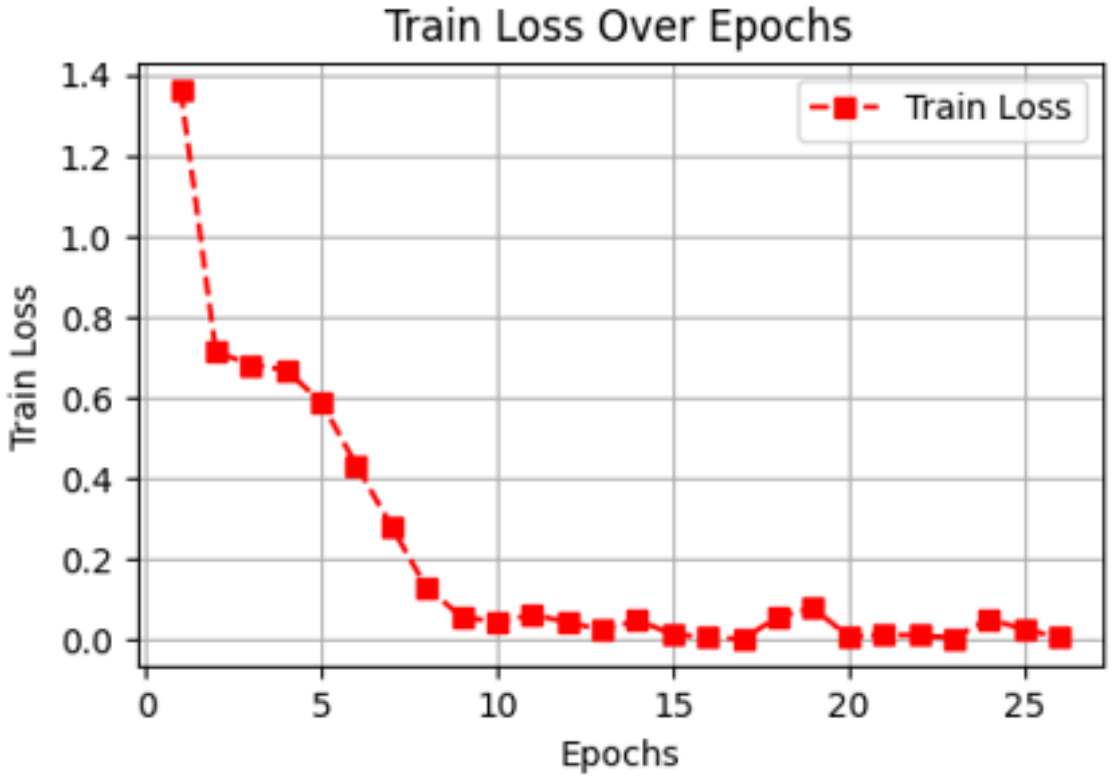

Figure 6.6: Recall Curve of HTNet

The training loss continuously decreases over the epochs. It shows that the model is effectively learning from the data. This confirms that the optimization process is working well, and the model is reducing prediction errors.

Confusion Matrix of HTNet

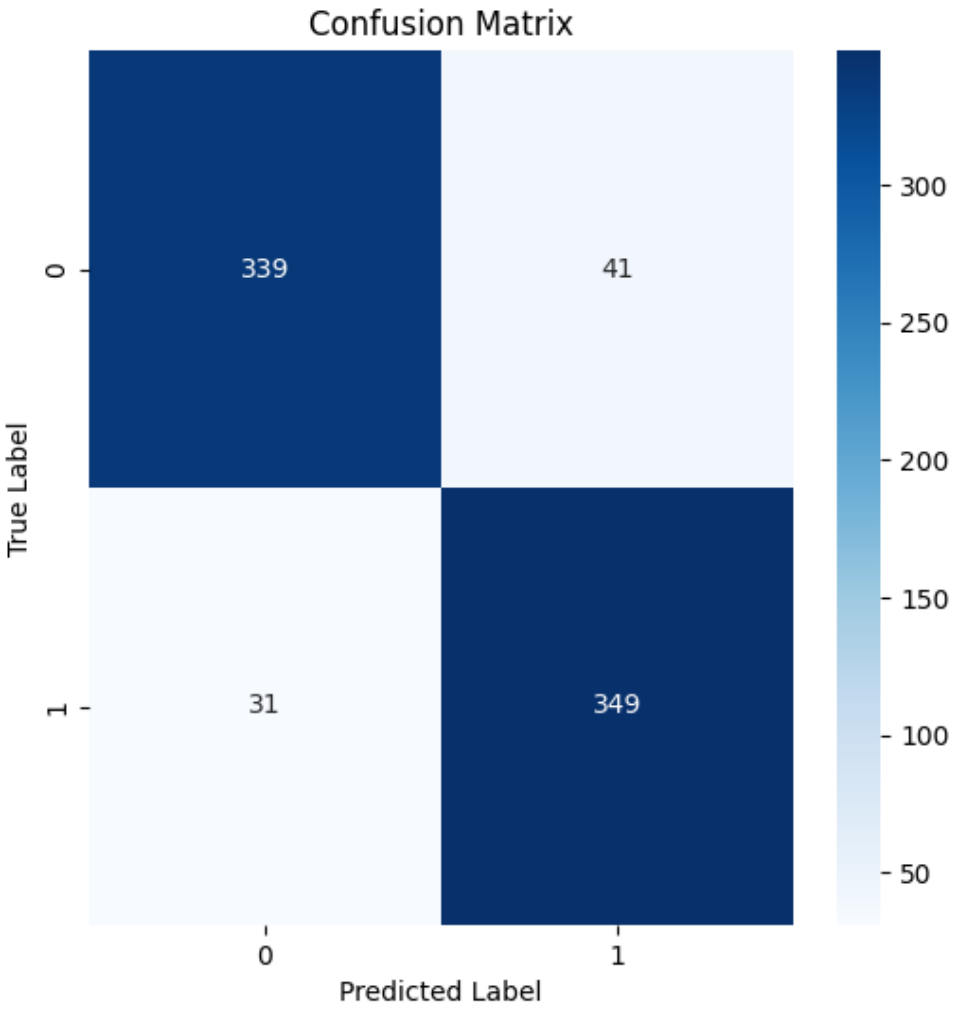

Figure 6.7: Confusion Matrix of HTNet Model

subsection*Confusion Matrix Analysis The confusion matrix provides the following classification results:

- **True Negatives (TN):** 349 (Correctly classified real images)
- **True Positives (TP):** 339 (Correctly classified deepfakes)
- **False Positives (FP):** 41 (Real images incorrectly classified as deepfakes)
- **False Negatives (FN):** 31 (Deepfakes incorrectly classified as real)

The relatively high true positive and true negative values indicate that the model performs well. However, reducing false negatives could further improve its ability to detect deepfakes more accurately.

Chapter 7

Comparative Analysis

7.1 Comparative Performance Analysis of Different Approaches for Deepfake Detection

Category	Approach 1 (FINAL)	Approach 2	Approach 3	Approach 4
Accuracy	91%-92%	95%-100%	98%-100%	57.4%-56.5%
F1 Score	0.80 - 0.91	0.96 - 1.00	0.95 - 1.00	0.96 - 0.99
Precision	0.80 - 0.91	0.91 - 1.00	1.00 - 1.00	0.93 - 0.96
Recall	0.80 - 0.90	0.965 - 1.00	0.990 - 1.00	0.997 - 0.996
Model Performance	Good	Poor	Poor	Very poor
Generalization	Good	Overfitting	Overfitting	Underfitting
Transfer Learning	Not Applied	Applied	Applied	Applied
Pre-trained Accuracy	N/A	86%	86%	86%
Inference Time (Optical Flow)	Very High	Very High	Moderate	Moderate
Regularization Techniques	Batch Dropout, Fusion Model	Batch Dropout, Fusion Model	Batch Dropout, Fusion Model	Batch Dropout, Fusion Model
Real Dataset	FaceForensics++ Original Sequence	CASME II, SMIC, SAMM Combined	CASME II, SMIC, SAMM Combined	CASME II, SMIC, SAMM Combined
Fake Dataset	FaceForensics++ Neural Textures	FaceForensics++ Neural Textures	FaceForensics++ Neural Textures	FaceForensics++ Neural Textures
Optical Flow	Using TVL1	Using TVL1	Real (Using Gunner Farnebäck) & Fake (Using TVL1)	Real(Gunner Farneback) & Fake(Raft)

Table 7.1: Comparative Analysis of Different Approaches

Approach 1: This is our final approach, which does not utilize transfer learning; therefore, no pre-trained weights were employed. Instead, the HTNet model was applied directly to our dataset. For the real dataset, optical flow features were extracted from the original FaceForensics++ sequences, while for the fake dataset, neural textures from FaceForensics++ were utilized. The dataset comprises approximately 1,900 samples, with an 80%–20% split for training and testing, respectively.

Optical flow was generated using TVL1 for both the real and fake datasets.

Approach 2: This approach utilizes transfer learning; hence, pre-trained weights were employed. The HTNet model was applied as described in the transfer learning section of Chapter 5 in our report. For the real dataset, optical flow features were extracted from a combination of the CASME II, SMIC, and SAMM datasets, while for the fake dataset, neural textures from FaceForensics++ were utilized. The dataset comprises approximately 1,900 samples, with an 80%–20% split for training and testing, respectively. This approach exhibits a clear overfitting issue. Optical flow was generated using TVL1 for both the real and fake datasets.

Approach 3: This approach also utilizes transfer learning, employing pre-trained weights. The HTNet model was applied as described in the transfer learning section of Chapter 5 in our report. For the real dataset, optical flow features were extracted from a combination of the CASME II, SMIC, and SAMM datasets, while for the fake dataset, neural textures from FaceForensics++ were utilized. The dataset comprises approximately 1,900 samples, with an 80%–20% split for training and testing, respectively. This approach also exhibits overfitting. Optical flow was generated using the Gunnar Farnebäck method for the real dataset, while TVL1 was used for the fake dataset.

Approach 4: This approach also utilizes transfer learning, employing pre-trained weights. The HTNet model was applied as described in the transfer learning section of Chapter 5 in our report. For the real dataset, optical flow features were extracted from a combination of the CASME II, SMIC, and SAMM datasets, while for the fake dataset, neural textures from FaceForensics++ were utilized. The dataset comprises approximately 27,370(half of which is for real and another half for fake) samples, with an 80%–20% split for training and testing, respectively. This approach exhibits underfitting. Optical flow was generated using the Gunnar Farnebäck method for the real dataset, while RAFT was used for the fake dataset.

7.1.1 Analysis

In this section, we analyze the performance of four distinct approaches for deepfake detection, evaluating key performance metrics such as accuracy, F1 score, precision, recall, model performance, and generalization ability. The approaches also differ in terms of dataset usage, transfer learning application, and optical flow generation methods.

Approach 1: No Transfer Learning
The performance of Approach 1 is deemed good, as it achieves an accuracy of 91%-92%, with F1 scores ranging from 0.80 to 0.91. Precision and recall are also within the range of 0.80 to 0.91 and 0.80 to 0.90, respectively. The model generalizes well and demonstrates good performance, without overfitting or underfitting. However, the inference time for optical flow generation is high due to the complexity of the process.

Approach 2: Transfer Learning with Overfitting
The performance of Approach 2 is relatively poor, as it exhibits overfitting. The accuracy is higher, ranging from 95% to 100%, with F1 scores between 0.96 and 1.00. Precision and recall are also high (ranging from 0.91 to 1.00 and 0.965 to 1.00, respectively). However, the model generalization is weak, leading to overfitting. The pre-trained accuracy is 86%, and the inference time is relatively low compared to other approaches.

Approach 3: Transfer Learning with Overfitting and Improved Optical Flow Generation
While Approach 3 shows slightly improved performance in terms of F1 scores, which range from 0.95 to 1.00, the model still suffers from overfitting, leading to poor generalization. Accuracy ranges from 98% to 100%, and precision and recall are also high (1.00 and 0.990 to 1.00, respectively). Like Approach 2, this approach also has a low inference time for optical flow generation, but the generalization ability remains a significant concern.

Approach 4: Transfer Learning with Underfitting
The performance of Approach 4 is considered very poor, with accuracy ranging from 57.4% to 56.5%, F1 scores between 0.96 and 0.99, and precision and recall ranging from 0.93 to 0.96 and 0.997 to 0.996, respectively. The model suffers from underfitting, leading to suboptimal performance. Inference times for optical flow generation are moderate, and regularization techniques such as batch dropout and fusion models are still employed.

Conclusion
From this analysis, it is clear that higher accuracy and performance metrics do not necessarily equate to better real-world applicability, especially when generalization is compromised. Approach 1, although not achieving the highest accuracy, provides a more reliable solution with good generalization and moderate inference time. On the other hand, Approaches 2 and 3 demonstrate high accuracy and performance but suffer from overfitting, rendering them less practical in dynamic, real-world settings. Approach 4, though using a larger dataset, suffers from underfitting, making it the least effective among the tested models. Future improvements should focus on optimizing model generalization, enhancing training with better regularization techniques, and considering inference time for real-time deepfake detection.

Chapter 8

Observation and Discussion

Observation

The confusion matrix shows that the model classifies positive and negative cases fairly accurately. However, there are some misclassifications that require attention:

- **False Negatives (31)** - The model sometimes misses positive cases, which is a risk for real-life applications that require precise detection.
- **False Positives (41)** - Some are wrongly classified as deepfakes.

Although the model is highly accurate, these errors suggest that it may not fully capture certain subtle aspects of the data, leading to occasional incorrect classifications.

Training Performance: A steady improvement is observed from the training metrics. Both training and validation loss decrease, indicating no major overfitting. The precision and recall values slightly fluctuate, suggesting that the model's accuracy in distinguishing true positives and avoiding false positives varies with different test samples. However, the convergence curve confirms that the model is learning effectively, with minimal gaps between training and validation losses. This signifies that the model generalizes well without overfitting.

While false negatives and false positives highlight areas for improvement, the model successfully identifies meaningful patterns. Refining decision boundaries could further improve classification accuracy. The model performs well, and its consistent loss reduction and stable evaluation metrics confirm a strong foundation for future optimizations.

Discussion

Detecting Deepfakes Using Micro-Expressions and HTNet

In this work, we propose an effective deepfake detection strategy leveraging **micro-expressions** and the **Hierarchical Transformer Network (HTNet)**. Traditional detection methods, such as pixel differences, unnatural blinking, and facial misalignment, are becoming ineffective as deepfake generation techniques advance. Our approach focuses on **micro-expressions**—brief, involuntary facial movements closely tied to human emotions. Since existing deepfake models struggle to replicate these expressions accurately, they provide a powerful cue for detection.

HTNet's Advantage in Identifying Deepfakes

HTNet was chosen for its ability to analyze subtle facial movements. Unlike conventional deepfake detection methods, which primarily examine pixel inconsistencies, HTNet studies the temporal dynamics of facial expressions by:

- Segmenting the face into distinct regions (eyes, lips, cheekbones).
- Analyzing localized micro-movements.
- Tracking how these regions move in coordination over time.

By applying hierarchical attention mechanisms, HTNet can focus on both **individual muscle movements** and **overall facial motion patterns**, making it highly effective in deepfake detection.

Baseline and Training Performance

The training results demonstrate that HTNet effectively learns and improves:

- Both **training and validation loss** steadily decrease.
- The model stabilizes with **accuracy between 90%-92%**, which is strong given the complexity of deepfake detection.
- **Precision and recall** vary slightly, indicating that HTNet performs better on some deepfakes than others.

These findings are further supported by the **convergence curve**, which shows continuous loss reduction without signs of overfitting. The small gap between training and validation losses confirms that HTNet generalizes well to unseen deepfake videos.

Why Micro-Expressions Help Detect Deepfakes

One key insight from this study is that deepfake models struggle to generate **authentic micro-expressions**. Unlike traditional methods that rely on visible artifacts, our approach leverages **facial muscle behavior under emotional stimuli**. Since micro-expressions are nearly impossible to fake, deepfake algorithms fail to replicate them accurately. HTNet enhances detection by:

- Performing **region-specific analysis**, dividing the face into multiple areas.
- Studying how facial muscles **work together naturally**.
- Identifying **unnatural movement patterns** indicative of deepfakes.

These aspects give HTNet an edge over conventional deepfake detection techniques that rely solely on pixel-based features.

Challenges and Areas for Improvement

Despite HTNet's strong performance, there are limitations:

- **False Negatives:** Advanced deepfake models can still evade detection.
- **Generalization Issues:** The model performs well on CASME II, SAMM, and FaceForensics++, but real-world deepfakes may differ significantly from training data.

To improve generalization and robustness, future research should:

- **Expand datasets** to include newer deepfake variations.
- **Fine-tune HTNet** to enhance its ability to detect emerging deepfake techniques.
- Explore **hybrid models** that combine micro-expression analysis with audio and lip-sync detection.

HTNet demonstrates significant potential in deepfake detection by leveraging micro-expressions. The results indicate that analyzing facial muscle dynamics provides a unique and effective approach for identifying manipulated content. Future improvements, such as incorporating more diverse datasets and hybrid detection methods, will further strengthen HTNet's ability to combat increasingly sophisticated deepfake techniques.

Chapter 9

Conclusion

This study presents an efficient approach for deepfake video detection through micro-expressions with an HTNet. In contrast toolder methods that look for mismatches in pixels, face alignment errors or unnatural blinking, this method searches for subtle, involuntary facial gestures that naturally accompany the expression of emotions. These micro-expressions are extremely difficult to fake,as they are processed by the human brain and last just under a second. Results indicate that HTNet can recognizeinconsistent expressions which in the end helps to know whether given video is real or fake. Its regularization allows itto accurately label most real and fake videos that solidifies it as a powerful tool for deepfake detection. But a few truly high-quality deepfakesstill evade detection, meaning that the model requires additional fine-tuning. A big benefit of this approach is that micro-expressions are closely tied to actual emotions,so AI-generated faces have far more difficulty generating these credibly. HTNet has some drawbacks, such asextremely realistic deepfakes that are difficult to detect, as well as an absence of diverse datasets to ascertain that the model works with different varieties of deepfake videos. In the future, the model can be made more robust by training it on more varyingdeepfake examples, enhancing it to detect low-level deepfake inconsistencies and integrating it with different approaches such as voice analysis and lip-sync analysis. Micro-expression deception thereforerepresents a new perspective and an important evaluation method for deepfake materials.

Bibliography

[1] S. Agarwal and L. R. Varshney, "Limits of deepfake detection: A robust estimation viewpoint," *CoRR*, vol. abs/1905.03493, 2019. arXiv: 1905.03493. [Online]. Available: http://arxiv.org/abs/1905.03493.

[2] I. Amerini, L. Galteri, R. Caldelli, and A. Del Bimbo, "Deepfake video detection through optical flow based cnn," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 1205–1207. DOI: 10.1109/ICCVW.2019.00152.

[3] S. L. Happy and A. Routray, "Fuzzy histogram of optical flow orientations for micro-expression recognition," *IEEE Transactions on Affective Computing*, vol. 10, no. 3, pp. 394–406, 2019. DOI: 10.1109/TAFFC.2017.2723386.

[4] J. Li, C. Soladié, R. Seguier, S.-J. Wang, and M. H. Yap, "Spotting micro-expressions on long videos sequences," May 2019, pp. 1–5. DOI: 10.1109/FG.2019.8756626.

[5] X. Li, T. Pfister, X. Huang, G. Zhao, and M. Pietikainen, "Spotting micro-expressions with a dual-inception network," *IEEE Transactions on Image Processing*, vol. 28, no. 12, pp. 6103–6111, 2019.

[6] S.-T. Liong, Y. Gan, J. See, H.-Q. Khor, and Y.-C. Huang, "Shallow triple stream three-dimensional cnn (ststnet) for micro-expression recognition," *arXiv preprint arXiv:1902.03634*, 2019. [Online]. Available: https://arxiv.org/abs/1902.03634.

[7] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, *Faceforensics++: Learning to detect manipulated facial images*, Jan. 2019. DOI: 10.48550/arXiv.1901.08971.

[8] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, *Faceforensics++: Learning to detect manipulated facial images*, Jan. 2019. DOI: 10.48550/arXiv.1901.08971.

[9] S. Agarwal, H. Farid, S.-N. Lim, and W. Scheirer, "Detecting deep-fake videos from phoneme-viseme mismatches," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 660–661. [Online]. Available: https://openaccess.thecvf.com/content_CVPRW_2020/papers/w39/Agarwal_Detecting_Deep-Fake_Videos_From_Phoneme-Viseme_Mismatches_CVPRW_2020_paper.pdf.

[10] T. Mittal, U. Bhattacharya, R. Chandra, A. Bera, and D. Manocha, "Emotions don't lie: An audio-visual deepfake detection method using affective cues," in *Proceedings of the 28th ACM International Conference on Multimedia*, ser. MM '20, Seattle, WA, USA: Association for Computing Machinery, 2020, pp. 2823–2832, ISBN: 9781450379885. DOI: 10.1145/3394171.3413570. [Online]. Available: https://doi.org/10.1145/3394171.3413570.

[11] R. Tolosana, R. Vera-Rodriguez, J. Fierrez, A. Morales, and J. Ortega-Garcia, "Deepfakes and beyond: A survey of face manipulation and fake detection," *Information Fusion*, vol. 64, pp. 131–148, 2020, ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2020.06.014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1566253520303110.

[12] L. Verdoliva, "Media forensics and deepfakes: An overview," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 910–932, 2020. DOI: 10.1109/JSTSP.2020.3002101.

[13] Z. Hanqing, T. Wei, W. Zhou, W. Zhang, D. Chen, and N. Yu, "Multi-attentional deepfake detection," Jun. 2021, pp. 2185–2194. DOI: 10.1109/CVPR46437.2021.00222.

[14] Y. Liu, P. Zhang, Y. Xu, and G. Zhao, "Main directional mean optical flow (mdmo) for micro-expression recognition," *arXiv preprint arXiv:2101.04838*, 2021. [Online]. Available: https://arxiv.org/abs/2101.04838.

[15] P. Yu, Z. Xia, J. Fei, and Y. Lu, "A survey on deepfake video detection," *IET Biom.*, vol. 10, pp. 607–624, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:234805191.

[16] L. Zhou, X. Shao, and Q. Mao, "A survey of micro-expression recognition," *Image and Vision Computing*, vol. 105, p. 104 043, 2021, ISSN: 0262-8856. DOI: https://doi.org/10.1016/j.imavis.2020.104043. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S026288562030175X.

[17] Y. Li, J. Wei, Y. Liu, J. Kauttonen, and G. Zhao, "Deep learning for micro-expression recognition: A survey," *IEEE Transactions on Affective Computing*, vol. PP, pp. 1–23, Oct. 2022. DOI: 10.1109/TAFFC.2022.3205170.

[18] S. Wang, S. Guan, H. Lin, J. Huang, F. Long, and J. Yao, "Micro-expression recognition based on optical flow and pcanet," *Sensors (Basel)*, vol. 22, no. 11, p. 4296, Jun. 2022. DOI: 10.3390/s22114296.

[19] H. Zhao, J. Li, Z. Sun, and T. Tan, "Multi-attentional deepfake detection," *Neurocomputing*, vol. 488, pp. 11–20, 2022.

[20] American Psychological Association, *Psycnet: Doi 10.1037/t27734-000*, Accessed: 2025-02-01, 2023. [Online]. Available: https://psycnet.apa.org/doiLanding?doi=10.1037%2Ft27734-000.

[21] B. Le, S. Tariq, A. Abuadbba, K. Moore, and S. Woo, "Why do facial deepfake detectors fail?" In *The 2nd Workshop on the security implications of Deepfakes and Cheapfakes*, ser. ASIA CCS '23, ACM, Jul. 2023. DOI: 10.1145/3595353.3595882. [Online]. Available: http://dx.doi.org/10.1145/3595353.3595882.

[22] X. Liao, Y. Wang, T. Wang, J. Hu, and X. Wu, "Famm: Facial muscle motions for detecting compressed deepfake videos over social networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 12, pp. 7236–7251, 2023. DOI: 10.1109/TCSVT.2023.3278310.

[23] C. Liu, H. Chen, T. Zhu, J. Zhang, and W. Zhou, "Making deepfakes more spurious: Evading deep face forgery detection via trace removal attack," *IEEE Transactions on Dependable and Secure Computing*, vol. 20, no. 6, pp. 5182–5196, 2023. DOI: 10.1109/TDSC.2023.3241604.

[24] H. Hao, S. Wang, H. Ben, Y. Hao, Y. Wang, and W. Wang, "Hierarchical space-time attention for micro-expression recognition," *arXiv preprint arXiv:2405.03202*, 2024. [Online]. Available: https://arxiv.org/abs/2405.03202.

[25] F. Hernandez, "Deepfake technology in corporate cybersecurity: Emerging threats and defense mechanisms," Dec. 2024.

[26] A. C. K, A. A. V, S. Das, and A. Das, " Latent Flow Diffusion for Deepfake Video Generation," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2024, pp. 3781–3790. DOI: 10.1109/CVPRW63382.2024.00382. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/CVPRW63382.2024.00382.

[27] A. Kaur, A. Hoshyar, V. Saikrishna, and S. Firmin, "Deepfake video detection: Challenges and opportunities," *Artificial Intelligence Review*, vol. 57, May 2024. DOI: 10.1007/s10462-024-10810-6.

[28] Y. Tu, J. Wu, L. Lu, S. Gao, and M. Li, "Face forgery video detection based on expression key sequences," *Journal of King Saud University - Computer and Information Sciences*, vol. 36, p. 102 142, Jul. 2024. DOI: 10.1016/j.jksuci.2024.102142.