National College of Ireland

Deepfake Detection Using a Lightweight Hybrid CNN-ViT Model for Low-Power Devices

MSc Research Project
M.Sc. AI for Business

Ajul Gopu Elayadath
Student ID: x23295872

School of Computing
National College of Ireland

Supervisor: Victor del Rosal

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Ajul Gopu Elayadath

Student ID: X23295872

Programme: M.Sc. AI for Business **Year:** 2024-2025

Module: MSc (Research) Practicum/Internship

Supervisor: Victor del Rosal

Submission Due Date: 11/08/2025

Project Title: Deepfake Detection Using a Lightweight Hybrid CNN-ViT Model for Low-Power Devices

Word Count: 6500 **Page Count** 20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Ajul Gopu Elayadath

Date: 10/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Deepfake Detection Using a Lightweight Hybrid CNN-ViT Model for Low-Power Devices

AJUL GOPU ELAYADATH
X23295872

Abstract

The rise in deepfake technology has introduced significant challenges in maintaining the authenticity of digital content. While high-performance deepfake models exist, they are typically large and resource-intensive, making them unsuitable for deployment on low-power devices such as smartphones and CPU-only systems. This research aims to address this gap by proposing a lightweight hybrid model that combines the strengths of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs).

To achieve this, a hybrid architecture combining MobileNetV2 (a compact CNN) for efficient local feature extraction and TinyViT (a small Vision Transformer) for capturing global context was designed, ensuring both performance and computational efficiency. The model is trained on preprocessed facial regions from deep-fake videos, which are extracted using MTCNN for face detection and OpenCV for frame extraction. Data augmentation, normalization, and frame-skipping techniques are employed to improve generalization and reduce redundancy. The training process involves freezing and unfreezing backbone layers, applying focal loss to handle class imbalance, and using early stopping to prevent overfitting. Post-training quantization is applied to compress the model, reducing its size and improving inference speed without significantly degrading accuracy.

The proposed system achieved 99.66% validation accuracy on FaceForensics++ and 69.0% accuracy on the unseen Celeb-DF v2 dataset. Quantisation reduced model size by 53% (32.8 MB to 15.4 MB) and improved fake recall from 0.50 to 0.55 and accuracy from 69% to 79% indicating enhanced sensitivity to manipulated content. The results show the feasibility of deploying the system in a resource-constrained environment.

Keywords: Deepfake detection, Convolutional Neural Networks, Vision Transformers, MobileNetV2, TinyViT, Model compression, Quantisation

1 Introduction

The recent progress in artificial intelligence has made it possible to create hyper-realistic but synthetic content, known as deepfakes. Deepfake media has rapidly become a critical concern for digital security, media integrity, and public trust. Initially powered by adversarial networks (GANs) and, more recently, by high-fidelity diffusion models, these techniques can convincingly alter or fabricate human faces, speech, and gestures in images and videos Yasir and Kim, (2025). As this threat continues to grow, particularly on social media and video-sharing platforms, the need for effective and accessible detection mechanisms has never been more urgent.

The initial deepfake detection method often relied on convolutional neural networks (CNNs) because of their ability to capture spatial features and facial artifacts. Although effective, the CNN-based model

lacks global context Dasgupta et al., (2025). Vision transformers (ViTs), on the other hand, are a promising alternative offering the ability to model long-range dependencies by processing images as sequences of patches (Khan et al., 2023). While ViT models offer high accuracy and performance, they require high computational resources, which hinder real-time deployment. Wang et al., (2024)

Existing deepfake detection solutions often combine the strengths of both CNN and ViT to mitigate their limitations as standalone backbones. These models offer high accuracy but require significant computational resources. These models typically required deployment on GPU-accelerated environments, making them impractical for real-world use on low-power or edge devices such as smartphones or CPU-only systems. This high resource requirement creates a deployment bottleneck that limits the reach of deepfake detection to end users.

To address this issue, the current research investigates how a lightweight hybrid model combining CNN and ViT architectures can be optimized for accurate and real-time deepfake detection on mobile and CPU-only systems. By leveraging the local feature extraction strengths of MobileNetV2 and the global context modeling capabilities of TinyViT, the project aims to design a model that is both computationally efficient and detection effective. Additionally, post-training quantization is applied to reduce the model's size and inference latency, facilitating deployment on edge platforms.

Research question

How can a lightweight hybrid CNN-ViT model be designed and optimized to achieve accurate and real-time deepfake detection on low-power devices such as mobile phones or CPU-only systems?

Objectives:

To address this question, the study has four main objectives:

- Investigate the current state-of-the-art in CNNs, ViTs, and hybrid models for deepfake detection.
- Design a hybrid lightweight model integrating MobileNetV2 (CNN) and TinyViT (ViT).
- Train and optimize the model using Focal Loss, class weighting, and early stopping on preprocessed facial data.
- Apply post-training quantization to compress the model and evaluate its performance.

The structure of this document is organized to reflect a logical progression. Section 2 contains a comprehensive literature review on CNN, ViT, and hybrid architectures for deepfake detection. Section 3 outlines the research methodology, including data processing, training strategy, and evaluation. Section 4 describes the model's architectural design and technical specifications. Section 5 presents the implementation details. Section 6 evaluates results and analyzes trade-offs in model compression. Section 7 concludes the report with key findings and future research directions.

2 Related Work

2.1 Deepfake Detection Using Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) have been widely used in early deepfake detection efforts due to their efficiency in capturing spatial features such as edges. Several models and architectures have been proposed to detect manipulations in still images and video sequences by identifying local artifacts or inconsistencies introduced by generative contents.

2.1.1 Spatial Feature Extraction Models

Earlier works on detecting deepfakes focused on images. Afchar et al., (2018) proposed MesoNet, a compact CNN model which is specially designed for deepfake detection using only four convolutional layers followed by dense layers with dropout and ReLU activation. The MesoNet model consists of two variants: Meso-4 and MesoInception-4, both of which emphasize mesoscopic feature extraction rather than fine-grained pixel details. The model demonstrated strong performance on the DeepfakeTIMIT dataset, achieving up to 95% accuracy. Despite its strong performance, the shallow architecture limits its ability to generalize across different datasets.

Similarly, N and Simon (2025) introduced DeepGuardNet, a novel CNN designed to focus on robustness and feature extraction efficiency. The model utilizes custom convolutional layers and regularization strategies to achieve notable performance on benchmark datasets, such as FaceForensics++. Despite its robustness, the model was not specifically optimized for low computational requirements, making it unsuitable for real-time or mobile deployment.

2.1.2 Temporal and Motion-Aware CNNs

Temporal modeling was introduced to capture motion inconsistencies that are not visible in frames. In a study conducted by Amerini et al. (2019), an optical flow-based CNN model was proposed to detect Inconsistencies In temporal motion patterns using VGG16 and ResNet50 backbones. The CNN model processed optical flow fields computed from consecutive video frames, enabling the detection of motion artifacts that are not visible in single frames. The model was evaluated on Face Forensics++ and achieved 93.6% and high AUC scores (~.91). Although the results are effective, the model requires accurate face tracking and introduces additional computational requirements.

In contrast, a study Saikia et al., (2022) expanded this idea, proposing a two-stage hybrid CNN-LSTM model that leverages optical flow features extracted from video frames. The CNN component encodes motion representations, which are then passed to the LSTM to model sequential dependencies. The models acquired 91.21% on Face Forensics and 79.49% on the Celeb-DF dataset. However, the two-stage structure introduces higher computational demands.

2.1.3 Lightweight CNN Optimization

A significant contribution to lightweight CNN optimization is provided by Noprisson et al. (2024), who focused on optimizing MobileNet, a lightweight CNN model for deepfake detection. The study introduced two variants, MobileReLUDr and MobileReLUL2. The dataset was tested on the Celeb-DF dataset and achieved an accuracy of 72.18% by integrating dropouts and L2 regularization to improve generalization. The results support the use of MobileNet as an effective lightweight backbone for deploying deepfake detection models, a particularly relevant approach.

The CNN models have served as a foundation for deep-fake detection for many years. The CNN models provide strong spatial and temporal feature extraction. Models like MobileNet can be used as an effective backbone; however, their inability to capture global dependencies and reduced robustness under domain shifts limit their standalone effectiveness, which sets the stage for Vision Transformers.

2.2 Deepfake Detection Using Vision Transformers (ViTs)

As deepfake content and techniques grow more sophisticated, the transition from the traditional convolutional neural networks (CNNs) has become evident. While the CNN model offers powerful localized feature extraction capabilities, they are less effective in modeling long-range dependencies.

Vision transformers (ViTs), on the other hand, process images as patches and apply self-attention, enabling them to capture global inconsistencies.

2.2.1 **Foundational ViT Architectures**

The early ViT-based system was the Vision Transformer for Deepfake Detection (ViT-DFD) proposed by Dosovitskiy et al. (2021). In their study, they adapted the original ViT framework to classify manipulated facial content by treating images as sequences of 16 × 16 patches and processing them through transformer blocks. The model outperformed traditional CNN models, such as those used in ImageNet, but requires large-scale, dataset-intensive computation.

2.2.2 Forgery-Focused Attention Mechanisms

To adapt ViT for forgery detection, Nguyen et al. (2024) introduced a transformer architecture augmented with vulnerability-guided attention mechanisms, named FakeFormer. Model learners focus attention on regions likely to contain manipulations by creating synthetic manipulation maps through a self-blending technique. The model also used L2-Att (Learned Local Attention), which dynamically adjusts attention scores based on forgery-prone zones during training. The model achieved over 98% accuracy across various datasets, including FaceForensics++ and Celeb-DF, despite showing peak performance when creating vulnerability maps, which introduced additional complexity.

In contrast, Chen et al. (2024) conducted a study to reduce data dependencies and maximize generalization, proposing GFF-ViT (Guided and Fused Frozen CLIP-ViT). This framework enhances the CLIP transformer with one module for guiding feature extraction toward deepfake artifacts and another for fusing multi-level features. The model achieved impressive results with minimal data. However, the guidance and fusion modules increase architectural complexity, making them unsuitable for a resource-constrained environment.

2.2.3 Lightweight and Efficient ViT Variants

Mishra et al., (2024) Their study demonstrated the efficiency of Swin Transformers for deepfake image detection. The Swin model uses a hierarchical representation and shifted windows to model both local and global dependencies efficiently. The study noted a high AUC of 0.99 and 97.91 % accuracy on Celeb-DF v2, and 95.71 % accuracy on FaceForensics++. Despite being more efficient, the model is heavier for lightweight deployment.

On the contrary, Wu et al. (2022) introduced TinyViT, a distilled transformer model designed to retain performance while significantly reducing computational complexity in a resource-constrained environment. TinyViT model retains high classification performance while remaining feasible for edge and mobile deployment. Although it was not initially designed for deepfake detection, its compact architecture makes it a compelling component in a hybrid or constrained-resource environment.

In Conclusion, Vision transformers offer robust generalization, global feature modeling, and adaptability across diverse manipulation techniques, making them ideal for deepfake detection tasks. The ViT models were able to address critical challenges in scalability and robustness, while efficient models like Swin Transformer and TinyViT address some constraints. However, the high computational demands and sensitivity to data open the way for the development of hybrid architectures that leverage the complementary strengths of CNNs for local texture encoding and ViTs for global reasoning.

2.3 Hybrid CNN-ViT Architectures for Deepfake Detection

As deepfake techniques are advancing rapidly, the traditional standalone Convolutional Neural Networks (CNNs), although powerful for extracting local spatial features, struggle to model global relationships with images and video frames. In contrast to this, Vision transformers (ViTs) utilise self-attention mechanisms to capture long-range dependencies. The hybrid CNN-ViT approach aims to utilize the strengths of CNNs for local feature analysis and ViTs for global modeling, offering better performance.

2.3.1 Dual-stream and parallel hybrids.

Although a study Ganguly et al., (2022) proposed ViXNet, a dual-stream architecture that combines an Xception CNN for spatial frequency analysis and vision transformers for patch-level reasoning. In this approach, both backbones operate in parallel, combining features before final classification. The model was evaluated on FaceForensics++ and Celeb-DF v2 datasets, achieving around 99% AUC, outperforming single backbone counterparts. However, the model faces generalization issues and uses computational complexity during inference.

2.3.2 Sequential hybrids.

In contrast, Wodajo (2021) introduced the Convolutional Vision Transformer (CViT), an architecture that begins with convolutional layers for low-level feature extraction, followed by transformer encoders for learning global relationships. The architecture reached an accuracy of 91.5% and an AUC of 0.91. The CViT model showed significant improvement when compared to traditional CNN models. However, the model relies on high-resolution data and large attention maps, making it less reliable for real-time deployment.

2.3.3 Attention-augmented hybrids.

(Wang *et al.*, 2023) In their study, they proposed the Deep Convolutional Pooling Transformer. The architecture integrates convolutional pooling blocks within transformer modules. The model utilises a re -attention mechanism that refines attention maps based on feature importance. The model was evaluated on FaceForensics++ and Celeb datasets and showcased state-of-the-art performance across all. However, the models introduce complexity with attention fusion and multi-scale pooling blocks, making the interface computationally costly.

2.3.4Ensemble-based hybrids.

In another comprehensive study Anan et al., (2025), a hybrid architecture was proposed that combines ResNet-34, DeiT (a ViT variant), and XceptionNet into an ensemble augmented by wavelet-domain frequency features. The ensemble model achieved an accuracy of 93.23% and 97.44% AUC on the DFWild-Cup dataset, outperforming traditional and standalone models. The model also displayed high robustness across manipulation types; however, the model's computational requirements reduce its practicality for real-time or edge deployment.

2.3.5 Lightweight hybrid variants.

For a more computationally restricted environment, Mehta and Rastegari (2021) proposed MobileViT, a lightweight transformer model that embeds CNN-based inductive biases into transformer blocks. Unlike the traditional CNN-ViT, the MobileViT integrates convolutional operations within the

transformer block, enhancing the contextual awareness of the model. Although the model is not evaluated on Deepfake datasets, it achieved state-of-the-art performance on ImageNet classification. The model is optimized for edge deployment, even though it lacks domain-specific adaptation.

On the other hand, Pan et al. (2022) proposed EdgeViT, a hybrid transformer architecture that utilizes spatial-aware CNN components in transformer tokenization, enabling vision processing on mobile devices. The model achieved a latency of 3.9 ms on mobile GPUs and yielded fine classification results. Although the model was not originally designed for deepfake detection, but the model aligns the need of real-world time interface.

To conclude, the hybrid CNN-ViT architecture offers a promising way for robust and efficient deepfake detection, combining local feature encoding with global context modeling. However, most designs are optimized for accuracy on high-performance hardware, with limited exploration of model compression and lightweight deployment.

2.4 Compression Techniques and Dataset Considerations

The deployment of deep fake detection model systems on low-powered devices like mobile phones or CPU-only systems introduces critical challenges to be addressed, like computational power, memory, and energy efficiency, without degrading the model performance. To address these challenges, various compression techniques have been explored.

In their study, Cheng et al. (2017) conducted a critical analysis of model compression techniques, highlighting pruning, quantization, knowledge distillation (KD), and low-rank factorization as primary methods to reduce model complexity without significant loss in accuracy. The study systematically explains each method and its underlying principles, providing comparisons of compression ratios and the speed gained. However, the study is a general purpose and does not focus on domain-specific findings. (Karathanasis et al., 2025) on their study provided a focused review on applying
The compression techniques for deepfake detection were examined, and it was found that 90% of the compression works without major accuracy changes when tested and trained on the same dataset.

Rossler *et al.*,(2019) through their study Introduced face forensic ++ making a significant advancement by providing manipulated videos across four different methods, including FaceSwap and DeepFakes. The dataset contains both raw and compressed video, allowing users to train and evaluate robust deepfake systems, making it a benchmark dataset for deepfake systems. However, the dataset still contains visible synthesis artifacts which may cause generalization issues.

In summary, the deepfake detection has evolved from the CNN-based models, which excel at local feature extraction but lack global context modelling, to ViTs, which capture long-range dependencies yet demand high computational resources. Hybrid CNN–ViT architectures combine these strengths, achieving superior accuracy and robustness across forgery types.
While most models are designed for high-performance hardware and cannot be deployed on lightweight architecture. Additionally, model compression techniques such as pruning, quantization, and knowledge distillation are underexplored in the deepfake domain.

3 Research Methodology

The section outlines the methodological framework adopted to design, develop and evaluate the lightweight hybrid Convolutional Neural Network–Vision Transformer (CNN–ViT) model for deepfake detection. This research employs a quantitative experimental methodology to develop a lightweight deepfake detection model that is both accurate and suitable for deployment on low-resource devices. The methodology is divided into five phases: data collection, preprocessing, model training, evaluation, model compression, and deployment.

3.1Data Collection and Preprocessing

The datasets used in this research comprise two widely accepted:

- **FaceForensics++** serves as the primary dataset for model training and validation.
- **Celeb-DF V2**: is used for testing and evaluating the cross-dataset generalisation as it contains more realistic manipulations.

These datasets are well-known in the deepfake research community for their diversity and quality of labeled video samples, representing both authentic and manipulated facial content. These datasets are considered Benchmark datasets for deepfake studies and projects.

3.1.1 Frame extraction and Face cropping

To optimize the computational efficiency. Frames were sampled at five-frame intervals using OpenCV. This approach helps reduce the redundancy between frames while preserving essential temporal variations. Each frame was then processed using MTCNN (Multi-task Cascaded Convolutional Networks) to localise and extract face regions from the frames.

3.1.2 Preprocessing Techniques:

Preprocessing steps were implemented to normalize and augment the data. All the cropped faces were resized to 224×224 pixels to match the input specifications of both MobileNetV2 and TinyViT backbones. The data is then normalized to a range between -1 and 1 with a mean of 0.5 and a standard deviation of 0.5 across RGB channels. To improve model robustness and stimulate real-world effects, a series of data augmentations was applied, including color jitter, horizontal flipping, Gaussian blur, and random rotation.

The Faceforensic++ dataset was divided into an 85% training and 15% validation split, while the Celeb dataset was used for the final evaluation. To handle class imbalance, class weights are computed from the inverse frequency of each class. These weights were applied long with a custom Focal Loss function, which was tuned using $\alpha = [2.0, 1.0]$ for fake and real classes, respectively.

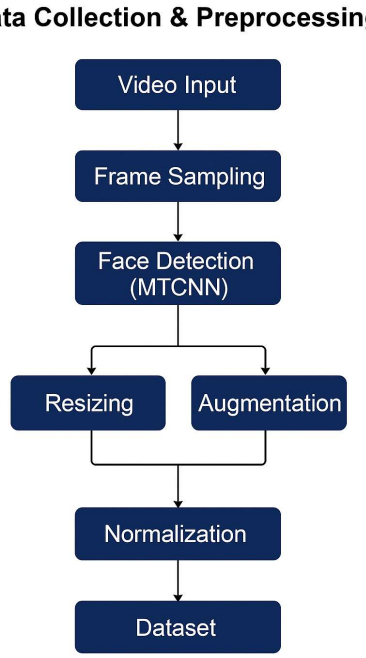

Figure 3-1data pipeline

3.2Model Training Methodology

The training part involved building a hybrid model that integrates the strengths of Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs). The MobileNetV2 backbone was employed to extract fine-grained local features, while the TinyViT component captured global dependencies through self-attention. And the output from both branches was concatenated before being passed to a fully connected classifier.

The training strategy was designed for both stability and efficiency. Training began with the CNN and ViT backbones frozen for five epochs, allowing the newly added fully connected layers to train effectively without interference. Following the initial phase, the backbones are unfrozen for fine-tuning using the Adam optimizer, starting with a learning rate of 1e-4, which was later reduced to 1e-5 after unfreezing. A **ReduceLROnPlateau** learning rate scheduler was used to adjust the learning rate based on the validation loss dynamically. To prevent overfitting, early stopping was employed with a patience parameter of five epochs, stopping the parameter when the validation loss is not improving.

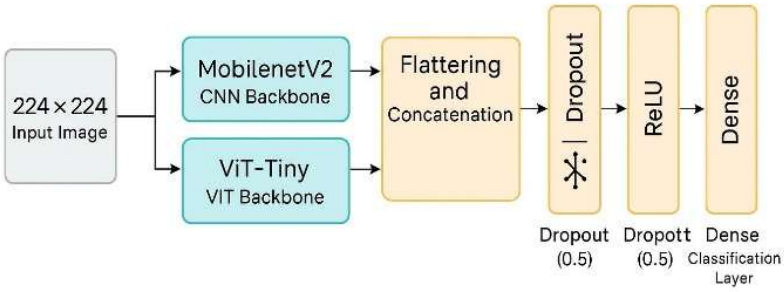

Figure 3-2 Hybrid Deepfake Detection Model

3.3Model Compression and Deployment

One of the core objectives of this research was to enable the deployment of the deepfake detection model on low-power devices, such as smartphones and CPU-only systems. To achieve this, the best-performing model from the training was compressed using post-training quantization (PTQ), a widely adopted technique for reducing model size and improving inference speed without requiring model retraining. Dynamic quantization fully connected linear layers, reducing both weights and activations from 32-bit floating point to 8-bit integers. To reduce the size of the model. This approach was chosen because it requires minimal changes to the model architecture and preserves most of the model's original accuracy.

The compressed model is then exported to TorchScript for deploying to edge devices. A Gradio web interface was developed, allowing users to upload videos, process frames in real-time, and receive predictions (Real or Fake) with confidence scores. The final application was deployed on Hugging Face Spaces, making it publicly accessible.

3.4Evaluation and Analysis

Model evaluation was conducted in two phases: an in-domain validation on the FaceForensics++ validation set. And a Cross-dataset test on Celeb-DF v2 data. The evaluation was conducted using a comprehensive suite of classification metrics. The values accuracy, precision, recall, and F1-score were calculated for each class to provide a balanced view of model performance, especially under class imbalance. The AUC-ROC (Area Under the Receiver Operating Characteristic Curve) was computed to assess the ability of the model to distinguish between the two classes, providing insight into the model's ability to distinguish fake samples from authentic ones.

4 Design Specification

The system is designed to integrate the advantages of convolutional networks and vision transformers into a single, lightweight model optimized for real-time deepfake detection on edge devices like smartphones or CPU-only systems.

4.1 Architectural Overview

The proposed solution is built on a hybrid architecture that combines a Convolutional Neural Network (CNN) backbone, MobileNetV2, with a Vision Transformer (ViT) backbone, TinyViT. MobileNetV2, A CNN-based model that efficiently extracts fine-grained local features using depthwise separable convolutions. It is specifically designed for mobile deployment due to its low parameter count and speed. TinyViT, on the other hand, is A compact Vision Transformer that captures global dependencies using multi-head self-attention. TinyViT has been optimized for speed and performance on low-powered devices.

The two backbones operate in parallel, each receiving the same 224×224 input image. The outputs of both backbones are flattened and concatenated. This combined output is passed through a Classification head, which consists of a Fully Connected Layer (512 neurons), ReLU activation, dropout (to prevent overfitting), and a final classification layer with softmax activation for binary classification. The dropout rate was tuned to 0.5 to balance generalization and performance.

This hybrid design addresses the weaknesses of each component. While CNNs excel at local feature extraction, they struggle with capturing global context. On the contrary, ViTs are excellent at modelling long-range dependencies but often require more data and are computationally heavier. The hybrid approach thus ensures robust performance while maintaining computational flexibility.

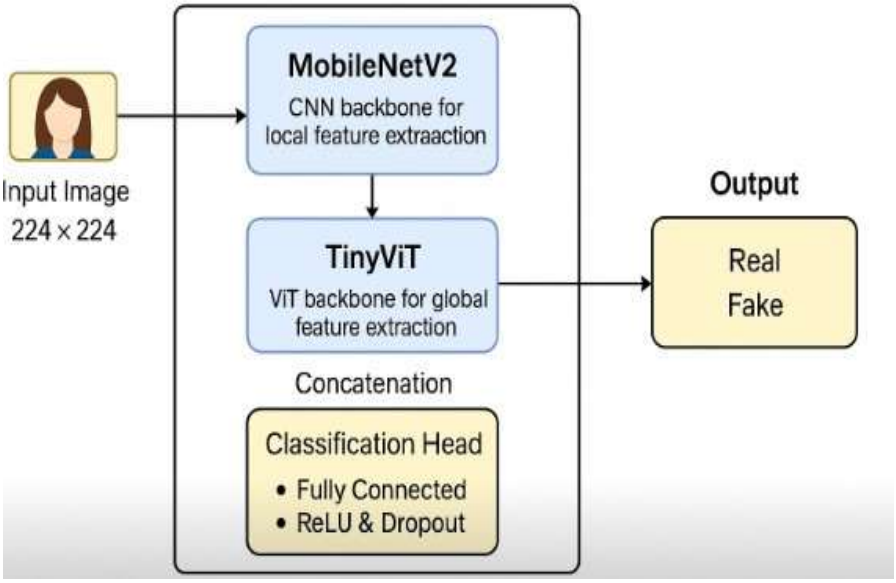

Figure 4-1 Model Architecture

4.2 Quantization-Friendly Architecture

To ensure the practical deployment of the model on edge devices such as smartphones and CPU-only systems, the architecture was explicitly designed to be quantization-friendly and lightweight. The

architecture utilized post-training dynamic quantization, which compresses the fully connected layers by converting them from 32-bit floating-point to 8-bit integer representations. The best-performing model is compressed.

During the architecture design phase, several considerations were taken to maximize compatibility with compression, such as implementing the classifier entirely using linear layers, which directly support quantization. Dropout was used to reduce overfitting, while avoiding batch normalization layers in the final stages, as they are not well-supported in post-training quantization pipelines. Both MobileNetV2 and TinyViT are independently modular and pre-trained, ensuring that quantization does not disrupt their core architecture.

4.3 Deployment-Optimised Output

Following the compression, the compressed mode is then exported into TorchScript and integrated into a GARDIO app, which allows users to upload videos and detect deepfakes in real time.

The final architecture ensures that the solution is not only academically robust but also deployment-ready, aligning with the project's aim to deliver a lightweight, efficient, and real-time accessible deepfake detection system.

5 Implementation

The implementation stage focused on developing a fully functional deepfake detection system using the proposed Hybrid CNN-ViT model, optimized for both acceptable accuracy and computational efficiency, to enable deployment on edge devices. This section outlines the complete process, starting from environment setup to deployment.

5.1Tools and Development Environment

All development and experimentation were conducted in the Google Colab Pro environment with access to Tesla T4 GPUs. The project was developed using Python 3.10 as the programming language and is implemented using the following core libraries and tools.

- PyTorch - model development and training, chosen for its flexibility in defining custom hybrid models.
- Torchvision – is used for image processing and augmentation.
- Timm - to pretrained Vision Transformer backbones efficiently.
- OpenCV is used for frame handling and frame extraction.
- facenet-pytorch – selected for its MTCNN implementation,
- matplotlib - for visualizing the training curve.
- sklearn – for computing evaluation metrics.
- Gradio – for creating an interactive, user-friendly web interface.
- Hugging Face Spaces – for public deployment without dedicated hosting infrastructure.

5.2Data Processing and Preprocessing Pipeline

Two benchmark datasets were selected to ensure strong in-domain accuracy and robust cross-dataset generalization.

- **FaceForensics**++ (FF++) - used for training and validation. This was chosen due to its variety of manipulation types.
- **Celeb-DF v2 –** used exclusively for testing to assess the model's robustness to unseen manipulations and higher video quality.

The preprocessing step is done in 2 phases. The first phase involved the following steps: -

1. Frame Sampling: A frame was extracted with five frame intervals using OpenCV to reduce redundancy.
2. Face Detection and Cropping: faces from each frame were detected using MTCNN, a robust face detector suitable for varied lighting and occlusion.
3. Cap on Face Samples: A maximum of **50** faces per video was enforced to balance representation and stored as JPG files.

The second phase involved data transformation.

Resizing: All the images were resized to **224 × 224 pixels to** match the input specification of MobileNetV2 and TinyViT.
Normalization: the pixel range was normalized to the range of **[-1,1].**
Random flipping: Introduces variability in face orientation, reducing bias.

Other arguments like color jitter and Gaussian blur are added to stimulate real world artifacts. To address the class imbalance in the dataset, class weight is calculated during this phase. The FF++ dataset was divided into **85%** training and **15%** validation subsets. Moreover, Data loaders were also implemented for effective loading of the dataset into the training process.

```
transform_train = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.RandomHorizontalFlip(p=0.5),
    transforms.ColorJitter(0.2, 0.2, 0.2, 0.1),
    transforms.RandomRotation(degrees=10),
    RandomApply([GaussianBlur(kernel_size=3)], p=0.3),  # simulate compression artifacts
    transforms.ToTensor(),
    transforms.Normalize([0.5] * 3, [0.5] * 3)
])

transform_test = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor(),
    transforms.Normalize([0.5] * 3, [0.5] * 3)
])
```

Figure 5-1 Preprocessing.py data augmentation

5.3Model Architecture and Training

The hybrid model was implemented in the hybrid_model_train.py script. It combines:

- MobileNetV2 for efficient spatial feature extraction
- TinyViT (a lightweight Vision Transformer) for attention-based global modeling

The features from both backbones were concatenated and passed through a fully connected head (512 neurons, ReLU, Dropout=0.5, and a SoftMax classifier). Initially, pretrained weights were frozen, and only the classifier was trained. After **five** epochs, the whole model was unfrozen and fine-tuned. The model was optimized using the Adam optimizer with a learning rate scheduler and early stopping. A custom implementation of Focal Loss with class balancing **(α = [2.0, 1.0], γ = 2.0)** was also used to address class imbalance, particularly enhancing detection of fake images. Validation metrics were logged, and the model with the best performance was saved as best_model.pth.

5.4Model Compression and TorchScript Export

The best performing weights **best_model.pth were compressed using post-training dynamic quantization using the compress_model.py script, resulting in a** reduction in model size and faster interface. Dynamic quantization was chosen over quantization-aware training (QAT) because it requires no retraining and is more resource-efficient.

The compression is followed by exporting the model to TorchScript. TorchScript was chosen to ensure portability and remove Python runtime dependencies. The TorchScript file is saved as **quantized_model_scripted.pt** making it ready for deployment.

5.5User Interface and Deployment

A real-time interface was developed using Gardio. The Gardio was chosen because it requires minimal setup and has straightforward requirements, providing a browser-based interface. The app accepts videos as input, extracts faces from frames, and classifies them into real or fake. Based on confidence and weighted voting. The app is deployed on Hugging Face Spaces to make it publicly available.

6 Results and Evaluation

This section presents a comprehensive evaluation of the proposed Hybrid CNN-ViT model for deepfake detection. The evaluation was conducted using the **FaceForensics++** dataset for training and validation, and the **Celeb-DF v2** dataset for testing. This evaluation section includes training behavior, performance on unseen data, ROC-AUC analysis, and the results of post-training dynamic evaluation.

6.1 Training and Validation Metrics

The Training process spanned **20 epochs.** During the initial stages of training, the CNN (MobileNetV2) and VIT (TinyVit) backbones were frozen, and after five epochs, they were unfrozen for end-to-end fine-tuning.

The impact of this methodology was immediately reflected in the learning curves (Figure 6.1). From Epochs 1 to 5, the training accuracy rose steadily, while the validation accuracy was around 79% and

the training loss decreased gradually. Once the backbones were unfrozen, the training and validation accuracy showed a sudden rise. Validation accuracy crossed 99% around epoch 10 and remained stable for the rest of the training process. While the training loss fell sharply.

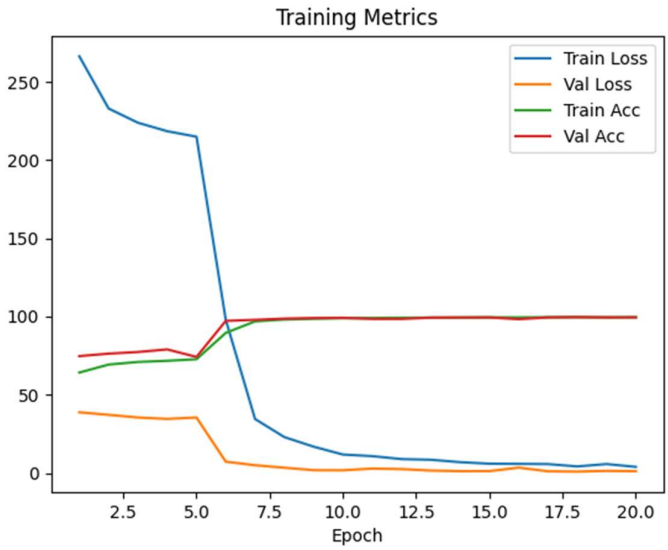

Figure 6-1Training and validation accuracy

From Figure 6.1, it is clear that the model has benefited from Focal Loss with class weighting, which addresses the dataset imbalance through the dropout mechanism and augmentation process.

However, the near-perfect validation accuracy on FaceForensics++ showcases the in-domain capabilities but also hints at a possible chance of dataset overfitting, making a cross-dataset evaluation essential.

6.2 Evaluation on Unseen Test Data (Celeb-DF v2)

Testing the model on the Celeb-DF v2 dataset served as a generalisation test, since the videos in this dataset are realistic and have fewer visible artefacts compared to the training dataset. The dataset contained 53,848 images extracted from videos with a balanced distribution of 24,584 fake and 29,264 real samples.

Class	Precision	Recall	F1-score	Support
Fake	0.74	0.50	0.60	24,584
Real	0.67	0.85	0.75	29,264
Accuracy			0.69	53,848
Macro Avg	0.71	0.68	0.68	53,848
Weighted Avg	0.71	0.69	0.68	53,848

Table 6-1 Classification Report

The model performs better in detecting real contents **(Recall = 0.85)** than fake content **(Recall = 0.50).** This lower fake recall suggests that the more convincing forged data in Celeb-DF v2 has been able to bypass the learned feature detectors, likely due to differences in the generative pipelines. The precision for fakes **0.74** indicates that when the model predicts that the data is fake, it would be true. A desirable property in forensic applications where false accusations must be avoided. A precision **of 0.7 across both classes shows that the model does not show excessive bias towards one class,** even if the fake recall is high.

This result highlights the challenges in cross-dataset generalization, despite having high validation accuracy; the model's accuracy had a notable drop when tested on the unseen dataset, especially in detecting sophisticated fake data.

6.3 ROC-AUC Analysis – original model

The ROC curve for the original model (Figure 6.3) achieved an AUC of **.79,** indicating that the model can distinguish fake from authentic images approximately **79%** of the time.

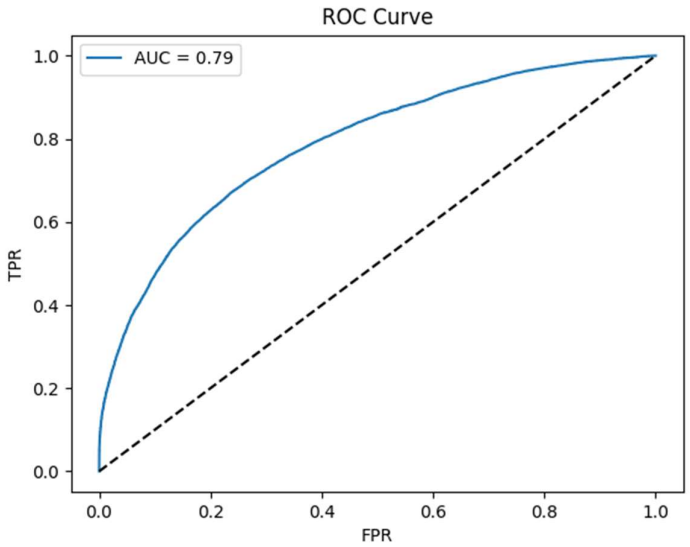

Figure 6-2 ROC curve for the original model

The shape of the curve shows room for improvement in reducing false negatives of the fake content.

6.4Quantized Model Evaluation

To optimize the model for deploying edge devices, dynamic quantization is applied to the trained model, lowering the model's weight to improve inference speed without compromising model accuracy. This evaluation was conducted to find out the effect of quantization on the model.

Class	Precision	Recall	F1-score	Support
Fake	0.74	0.55	0.63	24,584
Real	0.69	0.84	0.76	29,264
Accuracy			0.71	53,848
Macro Avg	0.71	0.69	0.70	53,848
Weighted Avg	0.71	0.71	0.70	53,848

Table 6-2 Quantized Model:

The quantized model maintains a level of performance that is comparable to the original model. The compressed model showed improved accuracy, ranging from 69% to 71%, **and** F1-score. Most importantly, the fake recall improved from **0.50 to 0.55**, indicating that the quantization-induced regularization has reduced the overfitting to the real Class. The results highlight that the compressed model performed better than the original model while being smaller in size.

6.5 ROC-AUC Performance (Quantized Model)

The ROC curve for the quantized model also achieved an AUC of 0.79 (Figure 6.3), which mirrors the original model, confirming that quantization preserves decision boundary quality.

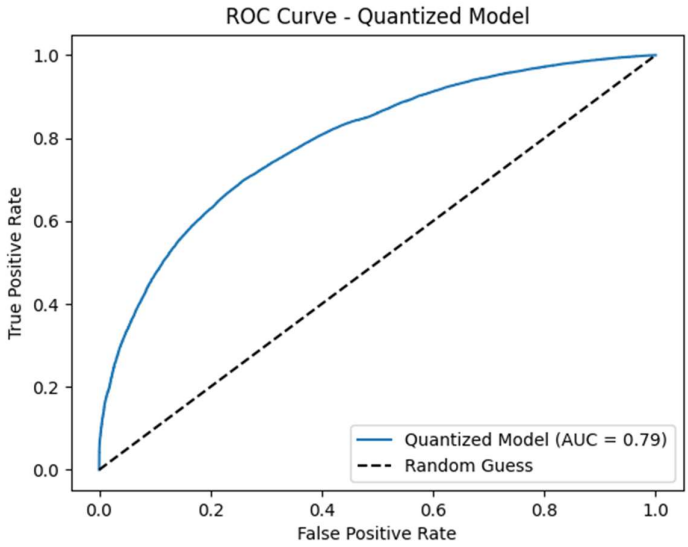

Figure 6-3 ROC curve for the quantified model.

6.5 Comparative Insights and Deployment Implications.

To assess the practical trade-off between accuracy and efficiency, a side-by-side comparison of the original trained model and the post-training quantified model was conducted.

Metric	Original Model	Quantized Model
Accuracy	0.69	**0.71**
Precision	0.71	0.71
Recall	0.68	**0.69**
F1-score	0.68	**0.70**
size	32.8 MB	15.4 MB

Table 6-3 Performance and Size Comparison

The Quantized model-maintained core classification performances, With a slight improvement in accuracy (+0.02), recall (+0.01), and F1-score (+0.02). These gains suggest that the quantized induced regularization has reduced overfitting. The rise in fake recall (+.05) indicates better sensitivity to manipulated content after quantization. And most importantly, the model's size reduces from **32.8 MB** to **15.4 MB**, representing **53%** reduction in size.

6.6 Critical Discussion of Findings

The performance of the proposed hybrid MobileNetV2–TinyViT architecture provides meaningful insights into the effectiveness and limitations of combining local and global feature extractors for deepfake detection.

The model achieved around **99%** validation accuracy on **the FaceForensics++ dataset,** showcasing the model's strong ability to detect manipulation in the training dataset. The performance hike after unfreezing **MobileNetV2 and TinyViT backbones** (Epoch 6) highlights the benefits of fine tuning. However, **an accuracy of 69% on the testing dataset highlighted** the cross-dataset challenges. Interestingly, the quantified model showed better performance when compared to the original model, showing an accuracy of 0.71% and most importantly, the fake recall value showed an improvement (0.50 to 0.55). These results suggest that the noise created by compressing helped the model to reduce overfitting to real class patterns. The size difference between the models, 32.8 MB to 15.4 MB, showcases the advancement toward the goal of
mobile and edge deployment.

7 Conclusion and Future Work

This project addressed the challenge of deploying deep-fake detection systems on low-powered edge devices by proposing a lightweight hybrid architecture that integrates MobileNetV2 for efficient local feature extraction and TinyViT for capturing global dependencies. The project was done through a carefully designed training strategy, which involved focal loss, class rebalancing, backbone freezing/unfreezing, and various data augmentations. The model achieved an in-domain validation accuracy of **99.66%** percentage on the Faceforensic++ dataset.

When evaluated on the unseen Celeb-DF v2 dataset, the model achieved **69.0%** accuracy, demonstrating moderate cross-dataset generalization. The application of post-training dynamic quantisation reduced model size by **53%** (**from 32.8 MB to 15.4 MB**). Additionally, it improved fake recall from 0.50 to 0.55 and accuracy from 69% to 71%, highlighting that compression can also act as a form of regularization. These results show the feasibility of deploying compressed lightweight deepfake detection systems on CPU-only and mobile environments without significant performance loss.

However, the model also lacks temporal flow modeling. The model operates solely on individual frames, without modelling temporal dependencies present in video content. And no fair assessment has been conducted on the model. And most importantly, cross-dataset generalization remains a challenge.

7.1 Future Work

While this project successfully delivers a lightweight and accurate hybrid deepfake detection system, several areas remain open for enhancement. Future work should focus on improving the model's generalization, fairness, interpretability and model performance.

- Future works should focus on integrating temporal modelling to extract motion artifacts (Amerini et al.,2019)
- Expand the training data by using more diverse datasets.
- Explore more compression techniques like pruning (Cheng *et al.*, 2017)
- Use interpretability tools to make the decision transparent.

8 Reference

Afchar, D., Nozick, V., Yamagishi, J. and Echizen, I. (2018) 'MesoNet: A compact facial video forgery detection network', *arXiv preprint*. Available at: https://arxiv.org/abs/1809.00888 (Accessed: 11 April 2025).

Amerini, I., Galteri, L., Caldelli, R. and Del Bimbo, A. (2019) 'Deepfake video detection through optical flow-based CNN', in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*. IEEE, pp. 1205–1211. Available at: https://doi.org/10.1109/ICCVW.2019.00156.

Anana, K., Bhattacharjee, A., Intesher, A., Islam, K., Fuad, A.A., Saha, U. and Imtiaz, H., 2025. Hybrid deepfake image detection: A comprehensive dataset-driven approach integrating convolutional and attention mechanisms with frequency domain features. *arXiv preprint* arXiv:2502.10682. Available at: https://doi.org/10.48550/arXiv.2502.10682

Celeb-DF Dataset, 2020. *Celeb-DF (v2)*. [online] Available at: https://github.com/yuezunli/celeb-deepfakeforensics?tab=readme-ov-file [Accessed 11 April 2025].

Chen, Y., Zhang, L. and Niu, Y., 2025. ForgeLens: Data-efficient forgery focus for generalisable forgery image detection. *arXiv preprint* arXiv:2408.13697v2. Available at: https://arxiv.og/abs/2408.13697v2

Cheng, Y., Wang, D., Zhou, P. and Zhang, T., 2017. A survey of model compression and acceleration for deep neural networks. *arXiv preprint* arXiv:1710.09282. Available at: https://doi.org/10.48550/arxiv.1710.09282

Dasgupta, S., Mason, J., Yuan, X., Odeyomi, O. and Roy, K., 2025. Enhancing deepfake detection using SE block attention with CNN. *arXiv preprint* arXiv:2506.10683. Available at: https://arxiv.org/abs/2506.10683v1

Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al., 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations (ICLR)*. Available at: https://arxiv.org/abs/2010.11929

FaceForensics++ Dataset, 2019. *FaceForensics++*. [online] Available at: https://github.com/ondyari/FaceForensics [Accessed 11 April 2025].

Ganguly, S., Mukherjee, S., Kundu, R., Roy, K. and Bandyopadhyay, S., 2022. ViXNet: Vision Transformer with Xception Network for deepfakes based video and image forgery detection. *Expert Systems with Applications*, 210, p.118423. Available at: https://doi.org/10.1016/j.eswa.2022.118423

Karathanasis, A., Violos, J., Kompatsiaris, I. and Papadopoulos, S., 2025. A brief review for compression and transfer learning techniques in deepfake detection. *arXiv preprint* arXiv:2504.21066v1. Available at: https://arxiv.org/abs/2504.21066v1

Khan, A., Rauf, Z., Sohail, A., Khan, A. R., Asif, H., Asif, A., & Farooq, U. (2023). *A survey of the vision transformers and their CNN-transformer-based variants*. Artificial Intelligence Review, 56(S3), 2917–2970. https://doi.org/10.1007/s10462-023-10595-0

Mehta, S. and Rastegari, M., 2021. MobileViT: Lightweight, general-purpose, and mobile-friendly vision transformer. *arXiv preprint* arXiv:2110.02178. Available at: https://doi.org/10.48550/arXiv.2110.02178

Mishra, S.R., Mohapatra, H. and Gourisaria, M.K., 2024. A robust approach for deepfake detection using SWIN Transformer. *Research Square*. Available at: https://doi.org/10.21203/rs.3.rs-4672886/v1

Nguyen, D., Astrid, M., Ghorbel, E. and Aouada, D., 2024. FakeFormer: Efficient vulnerability-driven transformers for generalisable deepfake detection. *arXiv preprint* arXiv:2410.21964. Available at: https://doi.org/10.48550/arXiv.2410.21964

Noprisson, H. *et al.* (2024) 'MOBILENET PERFORMANCE IMPROVEMENTS FOR DEEPFAKE IMAGE IDENTIFICATION USING ACTIVATION FUNCTION AND REGULARIZATION,' *JITK (Jurnal Ilmu Pengetahuan Dan Teknologi Komputer)*, 10(2), pp. 441–448. https://doi.org/10.33480/jitk.v10i2.5798.

N, A.D. and Simon, P. (2025) 'DeepGuardNet: A novel CNN architecture for DeepFake image detection', *Procedia Computer Science*, 258, pp. 811–818. Available at: https://doi.org/10.1016/j.procs.2025.04.313.

Pan, J., Zhuang, B., Liu, J., He, Z., Cai, J. and Lin, W., 2022. EdgeViTs: Competing lightweight CNNs on mobile devices with vision transformers. In *Lecture Notes in Computer Science*, pp.294–311. Available at: https://doi.org/10.1007/978-3-031-20083-0_18 .

Rössler, A., Cozzolino, D., Verdoliva, L., Riess, C., Thies, J. and Nießner, M., 2019. FaceForensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp.1–11. Available at: https://doi.org/10.1109/ICCV.2019.00009

Saikia, P., Dholaria, D., Yadav, P., Patel, V. and Roy, M. (2022) 'A hybrid CNN-LSTM model for video deepfake detection by leveraging optical flow features', in *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–8.

Wang, T., Huang, Z., Xu, Y., Guo, S., Li, B. and Zhang, Z., 2023. Deep convolutional pooling transformer for deepfake detection. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 19(6), pp.1–20. Available at: https://doi.org/10.1145/3588574

Wang, Z., Cheng, Z., Xiong, J., Xu, X., Li, T., Veeravalli, B., & Yang, X. (2024). A timely survey on Vision Transformer for Deepfake Detection. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2405.08463

Wodajo, D. and Atnafu, S., 2021. Deepfake video detection using Convolutional Vision Transformer. *arXiv preprint* arXiv:2102.11126. Available at: https://doi.org/10.48550/arXiv.2102.11126

Wu, Z., Han, K., Chen, Y., et al., 2022. TinyViT: Fast pretraining distillation for small vision transformers. In *Lecture Notes in Computer Science*, pp. 68–85. Available at: https://doi.org/10.1007/978-3-031-19803-8_5

Yasir, S. M., & Kim, H. (2025). Lightweight deepfake detection based on Multi-Feature Fusion. *Applied Sciences*, *15*(4), 1954. https://doi.org/10.3390/app15041954