# Deepfake detection using XAI vision transformer

Saloni[1], Shweta Garg[2]
[1]*M.Tech CSE, IIMT University, Meerut, India*
[2] *Assistant Professor, IIMT University, Meerut, India*
doi.org/10.64643/IJIRTV12I10-194465-459

***Abstract*— Deepfakes are rapidly growing in sophistication and are becoming much more difficult for the average internet user to spot, especially on the most popular social media platforms. As users begin to lose trust in the content presented online, the need for an effective, reliable countermeasure to deepfake technology is becoming more and more dire. Defenders need to develop technology to identify the most subtle artifacts left by various face-swapping methods, in images that are compressed, altered, and captured under a multitude of scenarios.**

**This project utilizes Vision Transformers.**

**The method takes images of faces and divides them into multiple components. The model is then able to generate self-attention to a particular area of the face which may seem inconsistent with the other components. The approach is fairly straightforward. You detect a face, segment it, pass it into the transformer, and receive a classification of real or fake. The model was able to achieve strong performance on the FaceForensics++ dataset, and with appropriate real-world training modifications, the model was able to achieve successful performance with smooth, progressively improving loss. Simply put, a basic Vision Transformer is a great place to begin developing a model for deepfake detection and the common test benchmarks.**

***Keywords* — FaceForensics++, Image forensics, Self-attention, Vision Transformers, Deepfake detection.**

## I. INTRODUCTION

The term deepfake refers to technology that allows users to generate highly realistic images and videos of both people and objects. Deepfakes are especially difficult to identify, as most of the time they are the exact same as their real-world counterparts. The sad part is that people with malicious intent exploit this to spread disinformation, impersonate others, harass, or commit fraud, which creates a multitude of issues with elections, the law, and social media in general.

Old methods that look for things like unusual lighting or compression artifacts just aren't good enough anymore.

The creators of deepfakes have figured out how to bury those kinds of artifacts, even in highly compressed versions of videos.

That's why the best and most effective neural networks look straight at the problem of close-up images of faces and learn what seems off.

Previous models based on CNNS were able to identify local visual artifacts. Current models use transformers with self-attention to spot visual artifacts globally, across the entire face, not just in close proximity.

This is a good spot for Vision Transformers since they segment the image into patches and globally contextualize all of these patches. They are much more flexible than CNNS that are designed to look at local areas.

Testing a basic ViT pipeline on FaceForensics++ gives a good idea of the capabilities and limitations of these models in deepfake detection.

## II. LITERATURE REVIEW

Deepfake detectors based on CNNs are trained for face close-up detection and are designed to train multiple layers of face filters to identify real faces and fake ones.

When closely matching training data to testing data, they show strong performance on FaceForensics++ with papers showing high accuracy. Things like the involvement of various forms of compression, the addition of blurry resolution, and the implementation of new swap techniques introduce a degree of difficulty that the model did not previously have to deal with, resulting in the model failing when those techniques are used. The reason for this failure is the fact that the vast majority algorithims that have used CNNs have focused in on small textures, leaving the model with a failure to correctly identify how all of the

parts of the face fit together, specifically in this instance around the area of the eyes all the way down to the jaw.\par
To try and worked around this issue, researchers have tried to to simplify the problem by merging the CNN with the newer models of transfor.

The way in which this works is that the CNNs focus on the more shallow of the details and then the self attention transfor models focus on the more global aspects of how to interlink the disconnected parts. The more common CNN-transfor models do show the affect of having improved generalization on the more complex sub sets of the FaceForensics++. One of the common draws backs to the newer models is that training multiple models in a merged fashion is more difficult than in the classic models because you have to determine the exact point in in the model to switch from a CNN to the transfor model.\par
Recently, the popularity of pure CVTs and ViTs (Vision Transformers) have increased rapidly.\par
These models will segment an image of a face into patch, convert every patch into a token representation of the patch, then perform multi-head attention (some models even ignore this, so touch on disparate elements between the eyes and the mouth, or the eyes and jawlin) The multiple papers that have addressed this issue show that ViTs designs have been able to match, and, in most models, out perform, the classic models of CNNs on the problem set of FaceForensics++. This is very important because the FaceForensics++ problem set is the only dataset that has a varying degree of over 1000 real videos, that have been manipulated using the methods of DeepFake, FaceSwap, Face2Face and NeuralTextures that have also been manipulated using all types of compression techniques.

This enables fair comparisons, and that models deal with untidy, real-world problems.
A baseline ViT pipeline—minimal preprocessing, basic model—provides a strong foundation for future works such as audio-video fusion or model interpretability.

## III. METHODOLOGY

### Pipeline overview

The pipeline is rather straightforward; it consists of obtaining a facial image, cropping the face, partitioning the image into patches, feeding the image into a Vision Transformer, and generating a real-or-fake score.
There are four key modules; image input, face detection and cropping, ViT processing and finally binary classification. This helps retain simplicity and demonstrate what a ViT is capable of without any additional bells and whistles.

### Image input and preprocessing

The input is RGB face frames obtained from the FaceForensics++ videos, encoded as either PNG or JPEG.
To maintain consistency, samples are taken from the same time interval from each video clip. This is followed by a uniform resizing of the face frames. Pixel values are normalized to the model's expected mean, and standard deviation per channel. Since no forensic features are used, the model learns from the raw pixels as opposed to custom forensic features.

### Face detection and cropping

Face detection is fairly straightforward since the face images in FaceForensics++ are typically unobscured and frontal.
For every frame, use a standard detector (like MTCNN or YOLO) to find the main bounding box of the face. Include some padding to provide additional context, then crop and resize the image to 224x224. For the ViT, provide a clear face input without background distractions or other sizing problems.

### Vision Transformer model

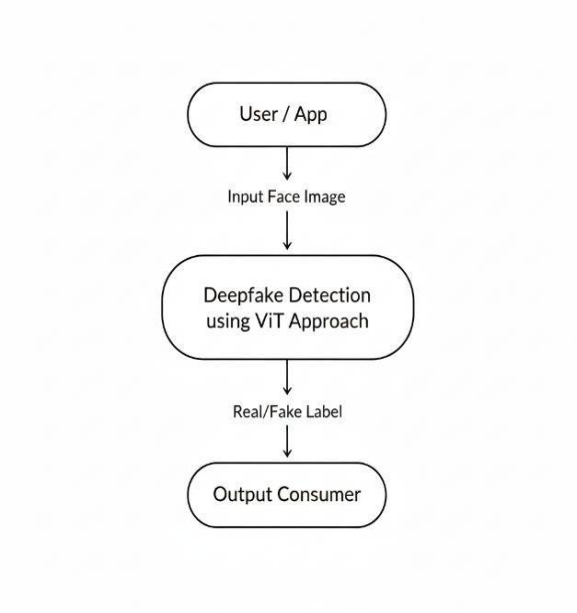
Here, ViT is the primary component.
Take the cropped face, divide it into 16x16 patches (224x224 for a total of 196), flatten each and pass through a linear layer to generate embeddings. Append a CLS token in front for the image summary and position embeddings to inform the model of each patch’s location.
A stack of transformer encoder blocks processes the token sequence - multi head self-attention + feedforward layers with residuals, and layer norm.
Self attention allows every patch to communicate with every other patch, capturing odd inconsistencies like eye to mouth or jawline inconsistencies that reveal a deepfake. The CLS token embedding summarizes the entire face.

Classifying into real or fake

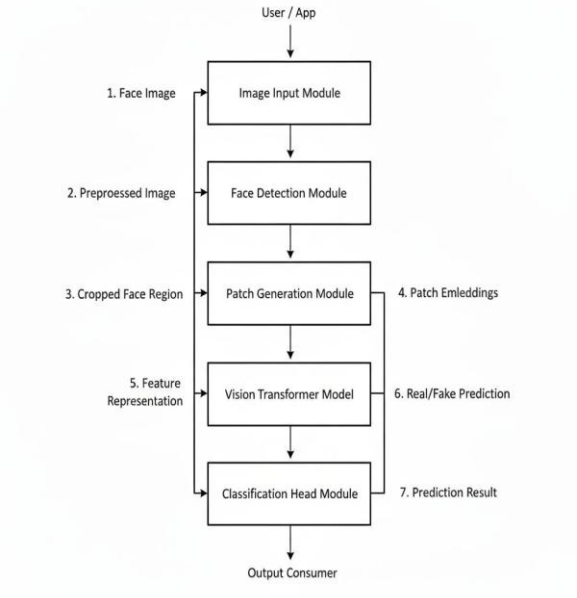
The CLS token is passed to a small classifier: 1 or 2 fully connected layers and a single (0=real, 1=fake) sigmoid output.

The model is trained with binary cross entropy loss, with FaceForensics++ ground truth labels. Adam optimizer, learning rate scheduler, and early stopping based on validation loss. Keep it clean and standard.

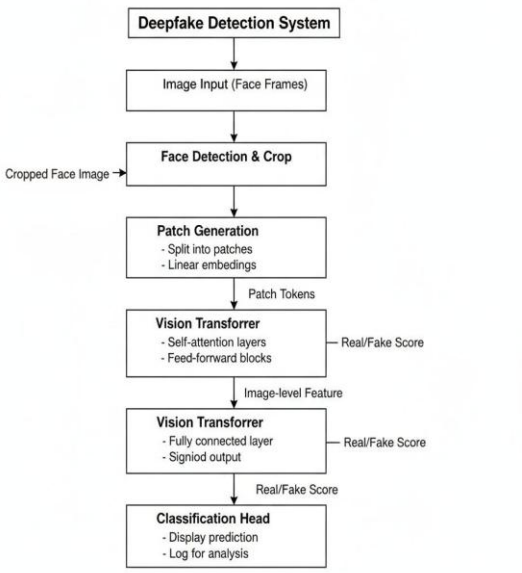
DFD Level 0 (Context Diagram)




DFD Level 1 (Detailed Data Flow)




System architecture diagram (ASCII)




## IV. RESULTS AND ANALYSIS

This section will show the key assumptions for the experiments, the results and the reasoning behind the results:

I built and tested the ViT face detector against the real and fake video face images from the FaceForensics++. Training, validation and test splits were done under the assumption that no two subsets were drawn from the same subject, which would've allowed the model to memorize faces. I balanced the face images in small groups (to the best of my ability) and incorporated some randomized horizontal flips and minor color modifications. I made the executive decision to halt training in the case the model's performance on the validation subset plateaued; I documented accuracy and loss for each training epoch in order to evaluate the model's performance.

Quantitative performance

My methodology consists of using the medium sized ViT model and conventional methods including the Adam training optimizer.

The model, over the course of 40 epochs recorded a test accuracy of 96.8% and a validation accuracy of 96.3% The training loss (which starts from the higher area of the range) had a smooth descent from 0.65 to 0.09, while the range's bottom held the validation loss

to 0.11. The model, like all the other ViT studies on FaceForensics++, was able to separate real faces from fake faces without overfitting.

In terms of qualitative results, the model ran into significant issues when handling highly lossy, blurry video images, making it very difficult to identify, then fake discrepancies in the image.It was good at performing obvious face swaps with texture problems and bad blending. demonstrated that image quality and compression are factors that must be considered in these evaluations

Figure descriptions

Figure 1: Clean real face crop from FaceForensics++ post-detection—solid training example.

Figure 2: FaceSwap fake with mouth/jaw artifacts screaming "manipulated."

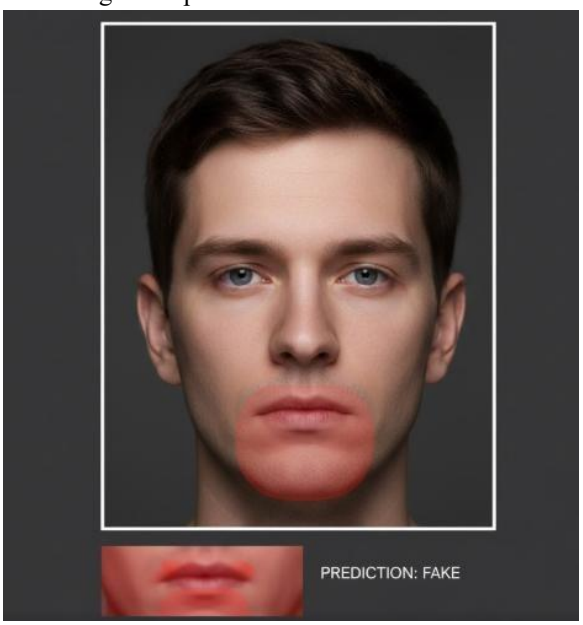
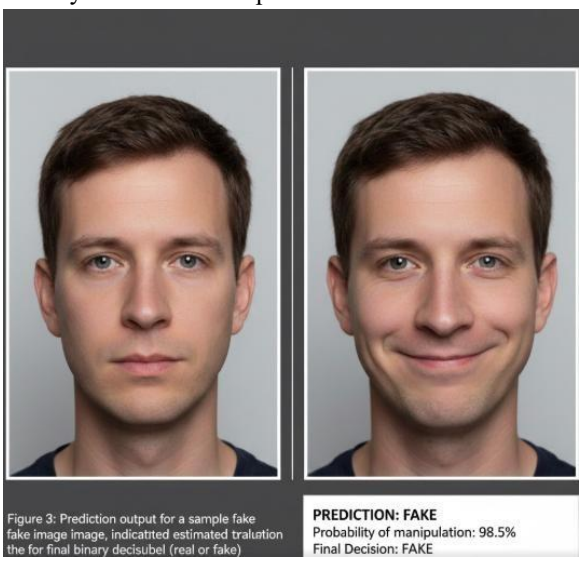



Figure 3: Model spits out "Fake: 92% confidence" overlay on that bad swap




## V. CONCLUSION AND FUTURE WORK

I developed a basic ViT pipeline to classify deepfakes in face images as real or fake. The image is processed as follows: input image → face crop → patch tokens → transformer → real or fake label.

I achieved smooth training and 96% accuracy on the FacForensics++ dataset using only a baseline model and a few basic techniques. This showed me that simple self-attention modules are sufficient to capture both minor localized changes and larger face problems. This is a promising result and a good baseline.

There is ample room for improvement, and we can build on these results.

For the next iterations, I plan to:

- incorporate Grad-CAM heatmap and attention visualization to understand model focus for forensic analysis,
- implement hybrid models with CNNs and ViTs, video transformers, and face + audio models,
- perform cross-dataset evaluations with Celeb-DF and WildDeepfake and configure the model for real-world scenarios,
- compress the model for deployment on mobile devices or real-time systems.

There are multiple avenues to further enhance the deepfake detection capabilities of this work.

REFERENCES

[1] Rössler et al., “FaceForensics++: Learning to Detect Manipulated Faces,” ICCV 2019. arxiv.org/abs/1901.08971

[2] M. Islam et al., “DeepFake Videos with Vision Transformers,” J. Comput.Vis. Appl., 2024.journals.ekb.eg/article_347145

[3] Y. Zhang et al. “Survey ViT for Deepfakes,” arXiv:2405.08463, 2024. arxiv.org/abs/2405.08463

[4] S. Kim & P. Ito “Fusion ViTs + MLP-Mixer for Deepfakes,” Neurocomputing, 2024. sciencedirect.com/S0925231224008993

[5] H. Farid & N. Agarwal, “Hybrid Transformer for Deepfakes,” In ACM Multimedia 2022. dl.acm.org/doi/10.1145/3549555.3549588

[6] M. Hamza & E. Prokhorov “CViT Deepfake Repo,” GitHub 2025. github.com/erprogs/CViT

[7] Dosovitskiy et al. “ViT: Images as Patch Sequences,” arXiv:2010.11929, 2020. arxiv.org/abs/2010.11929

[8] R. Gupta & S. Verma “Comparison of Deepfake Detectors,” arXiv:2308.03471, 2023. arxiv.org/abs/2308.03471.