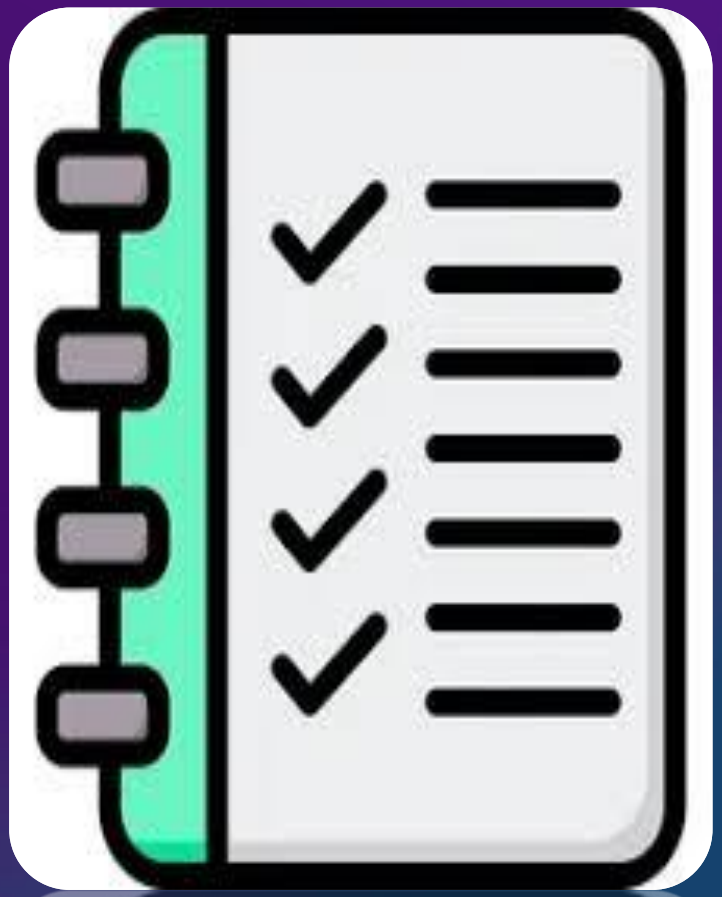
Hybrid RNN to Differentiate Deepfakes and Real Images/Videos

PROJECT BY:

Piyush S Wandile & Debaditya Paul

AGENDA



1. Need to detect Deepfakes
2. Datasets
3. Models
4. Comparison with existing tools
5. Prediction results
6. Better Model
7. Future Scope

Need to detect deepfakes

1

AI-generated media risks undermining **public confidence** in digital content, with synthetic images now achieving photorealism that challenges human discernment

2

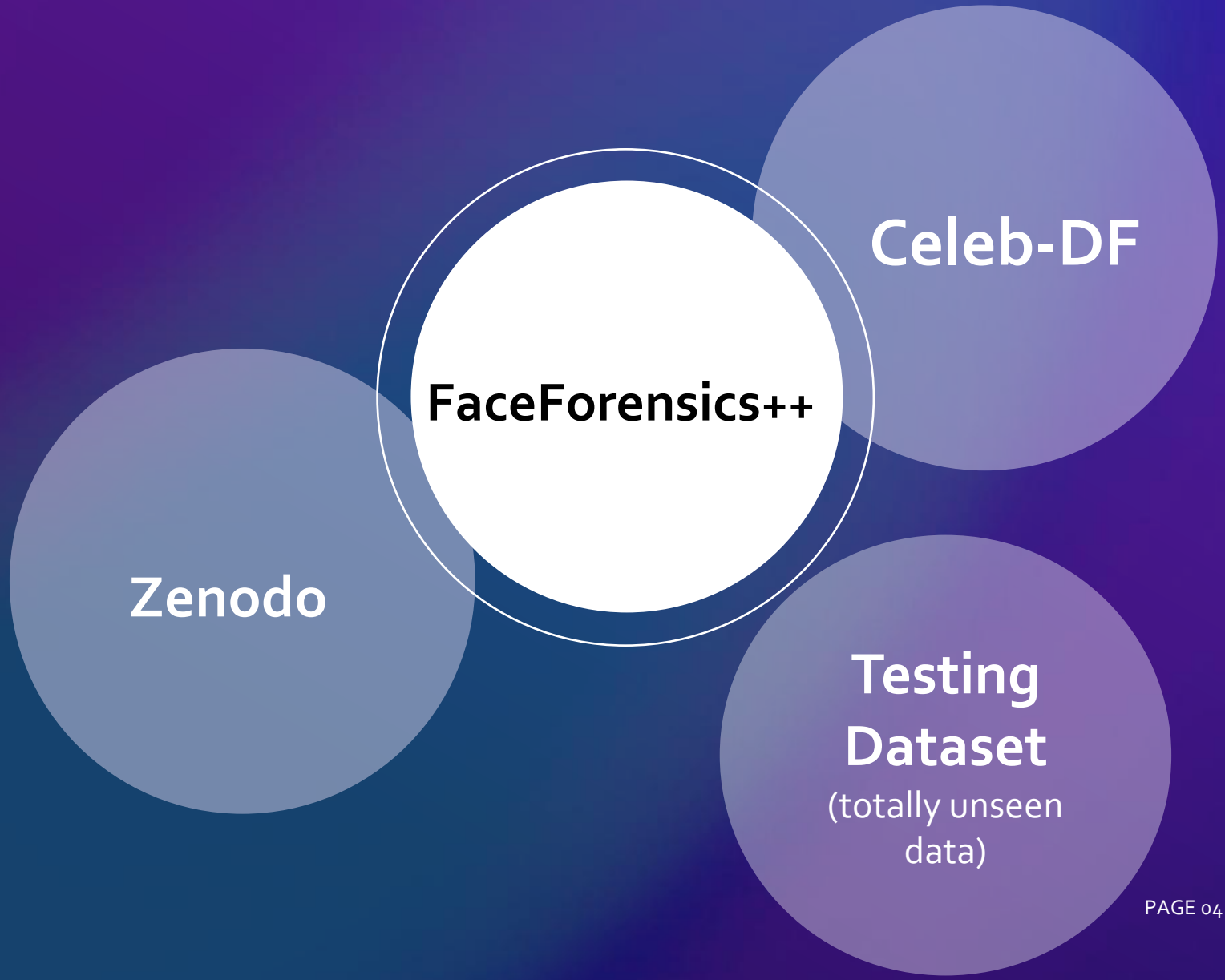
High-profile cases like AI-generated images of Pope Francis and deepfake audio of **political figures** demonstrate synthetic media's capacity to spread **false narratives**

3

Image generators often perpetuate occupational and gender stereotypes from training data, disproportionately sexualizing women and underrepresenting minorities in professional roles.

DATASETS

- These datasets would be used to train the model to differentiate between real and fake images or videos.
- **Stratified Sampling** will be used to balance the combined datasets.



Zenodo
Image Dataset




Source:
https://zenodo.org/records/5528418#.YpdlS2hBzDd

- **Rich Annotations:** Provides detailed, face-wise annotations to facilitate both forgery detection and segmentation tasks.
- **Challenging Scenarios:** Captures a variety of complex, in-the-wild conditions to test and improve the robustness of detection algorithms.
- **Sample Space:** Includes multiple subsets such as training, validation, and test sets with a total of **~95,000** real and **~95,000** fake images.

FaceForensics++ Video Dataset

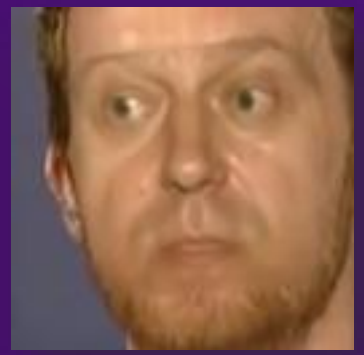
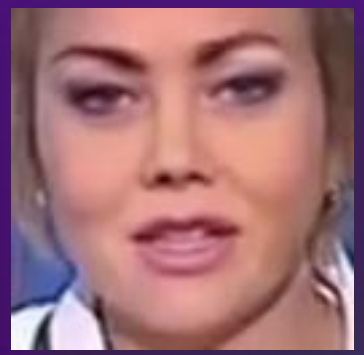
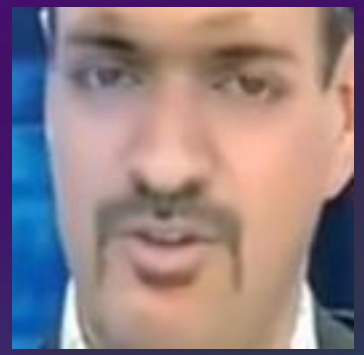
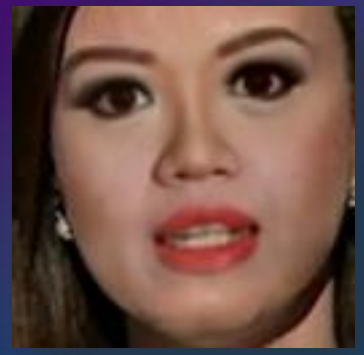
Source:
https://www.kaggle.com/datasets/hungle3401/faceforensics

FaceForensics++ is a forensics dataset consisting of **1000 original video** sequences that have been manipulated with four automated face manipulation methods: Deepfakes, Face2Face, FaceSwap and NeuralTextures.

The data has been sourced from 977 youtube videos and all videos contain a trackable mostly frontal face without occlusions which enables automated tampering methods to generate realistic forgeries.

FaceForensics++ is a large-scale dataset and benchmark designed for detecting manipulated facial images. It addresses the growing societal concerns over synthetic image generation and manipulation, such as the spread of misinformation and erosion of trust in digital content.

Celeb-DF Video Dataset

Source:
https://www.kaggle.com/datasets/nanduncs/1000-videos-split

- **Composition:** Comprises **590 real videos** sourced from YouTube, featuring celebrities of diverse ages, ethnicities, and genders, along with **5,639 corresponding DeepFake videos.**

- **Visual Quality:** The DeepFake videos are generated using an improved synthesis process, resulting in high visual quality comparable to DeepFakes found online.

- **Diversity:** The real videos feature subjects from various age groups, ethnic backgrounds, and genders, ensuring a broad representation.

- **Purpose:** Serves as a challenging benchmark for assessing the effectiveness of DeepFake detection methods, reflecting the complexities of real-world scenarios.

Testing Dataset

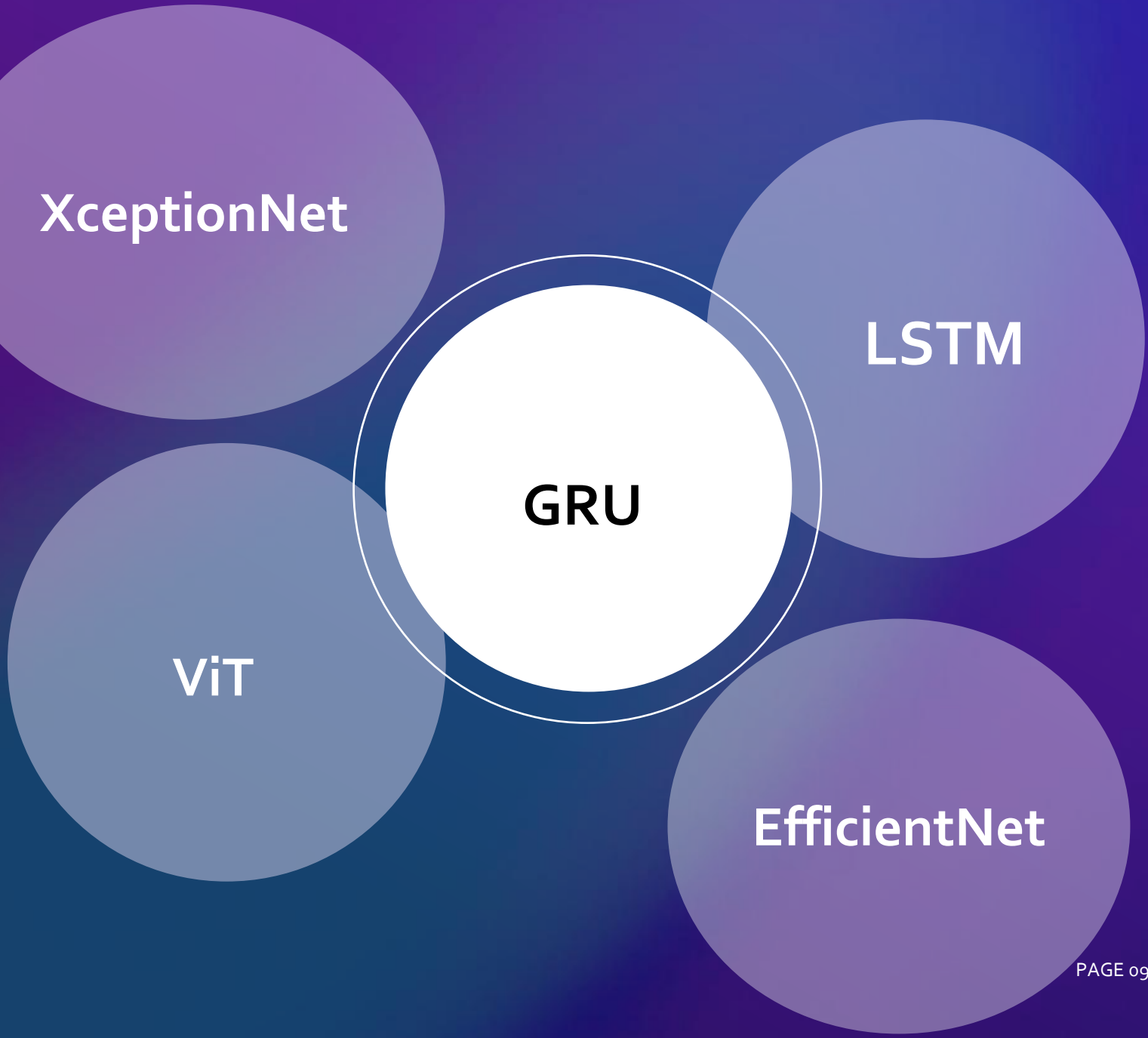



Source:
https://www.kaggle.com/datasets/xhlulu/140k-real-and-fake-faces/data

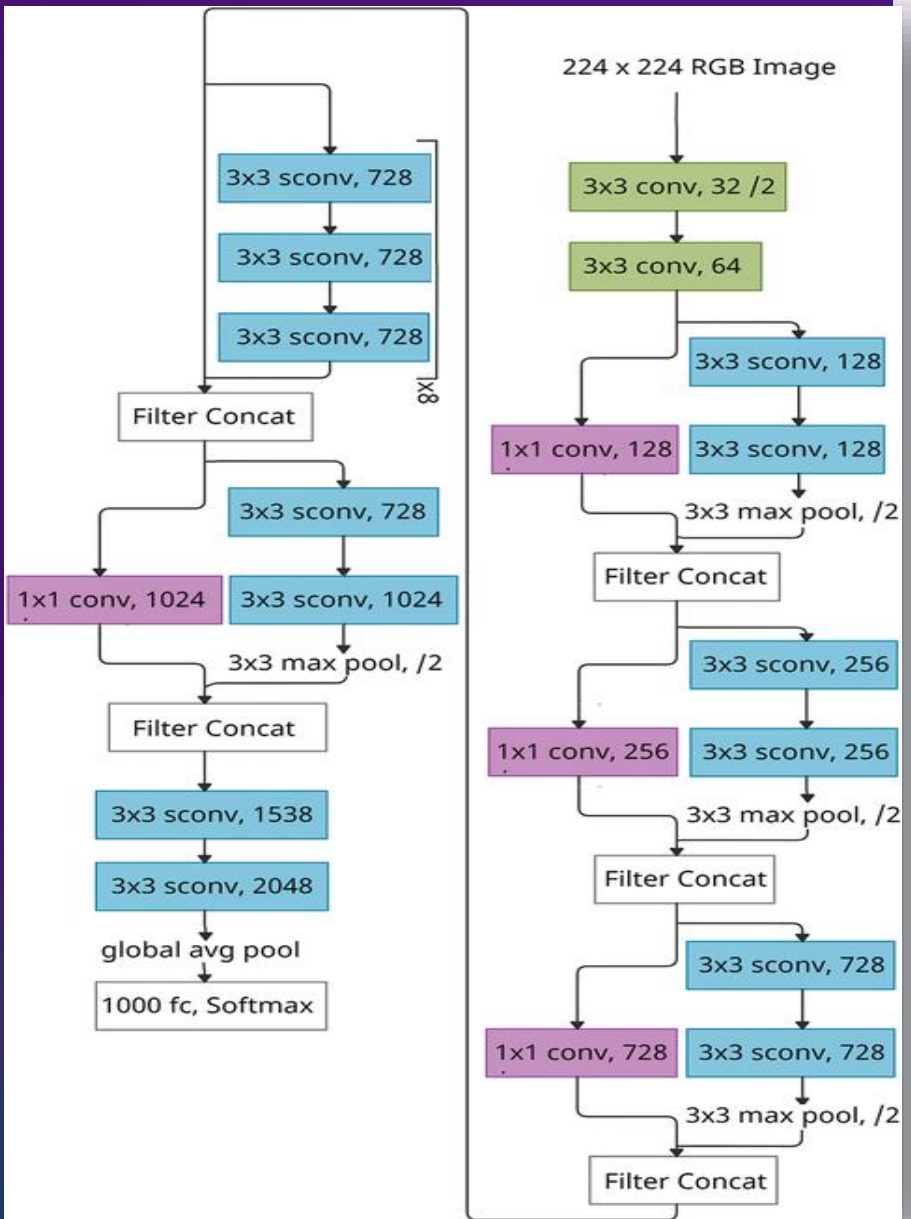
- **Composition:** Comprises **70,000** real face images sourced from the Flickr-Faces-HQ (FFHQ) dataset (collected by Nvidia). Includes **70,000** synthetic face images generated using StyleGAN (sampled from the "1 Million Fake Faces" dataset).
- **Advancement Over Previous Version:** Unlike earlier datasets that focused only on limited fake samples, this version provides a large-scale, balanced collection (140k images) for robust model training. The fake faces are derived from a high-quality StyleGAN generation process, improving realism over older GAN-based datasets.
- **Visual Quality:** All images are resized to 256x256 pixels, ensuring uniformity. The fake faces exhibit varying levels of realism, with some nearly indistinguishable from real ones (similar to high-quality DeepFakes).
- **Diversity:** Real faces cover a wide range of ages, ethnicities, and lighting conditions (from FFHQ). Fake faces include different synthesis difficulties (though not explicitly labeled in this Kaggle version).
- **Purpose:** Designed for training and evaluating deepfake detection models on static images. Useful for GAN-generated image classification, facial forgery detection, and computer vision research.

MODELS

- XceptionNet
- LSTM (Long Short-Term Memory)
- GRU (Gated Recurrent Unit)
- ViT (Vision Transform)
- EfficientNet



XCEPTIONNET



Architecture

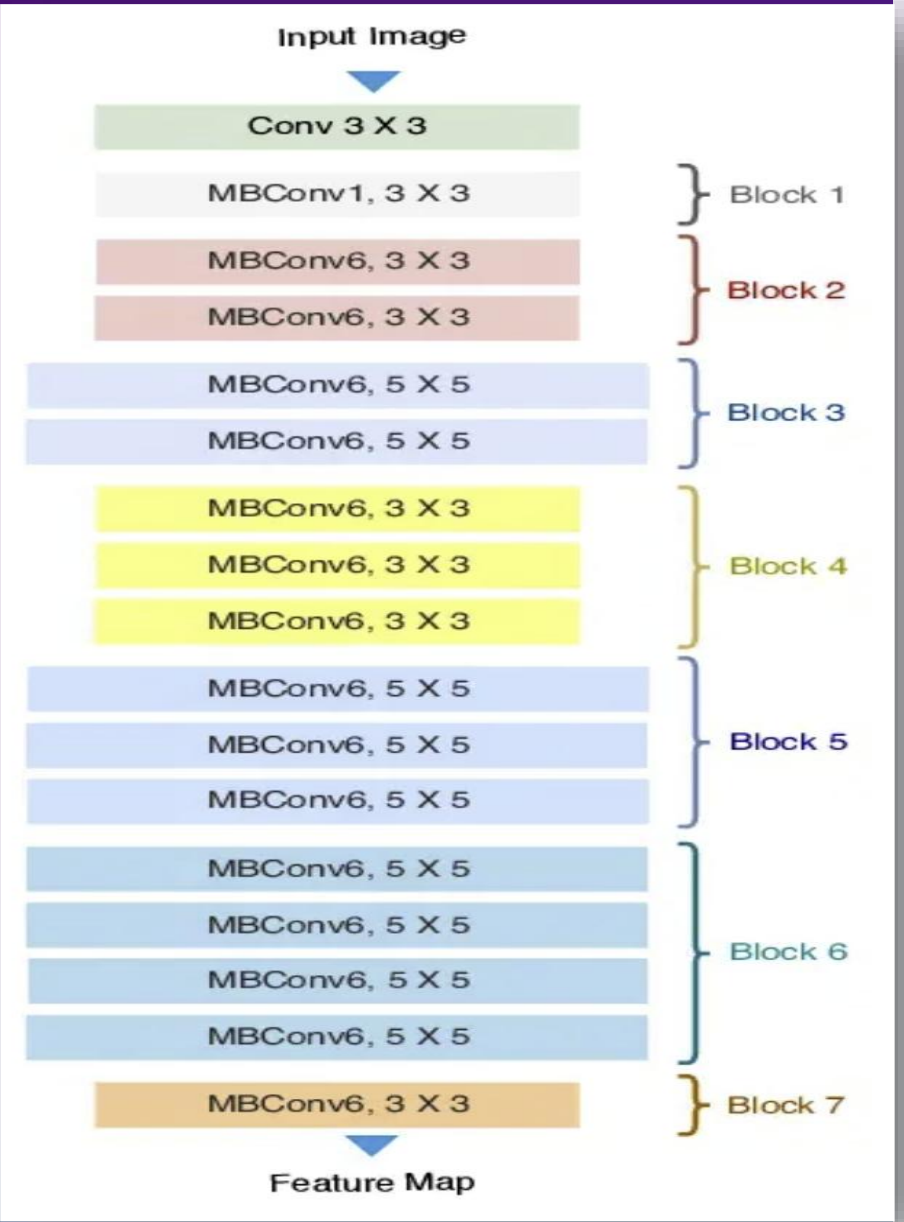
XceptionNet is a **36-layer deep CNN** that replaces Inception modules with depthwise separable convolutions—first applying a pointwise (1X1) convolution, then a depthwise (per-channel) convolution—allowing more efficient cross-channel and spatial correlation learning

Deepfake Detection Use

Pre-trained XceptionNet models are fine-tuned on deepfake frame datasets, extracting high-level facial feature maps. Their robust feature extraction and stable training have delivered accuracies up to 99.65% on large-scale video benchmarks

- Deepfakes often have visual inconsistencies like unnatural textures, blurring, or artifacts.
- ViTs are excellent at capturing global patterns and relationships, making them ideal for spotting such artifacts in individual frames.

EFFICIENTNET



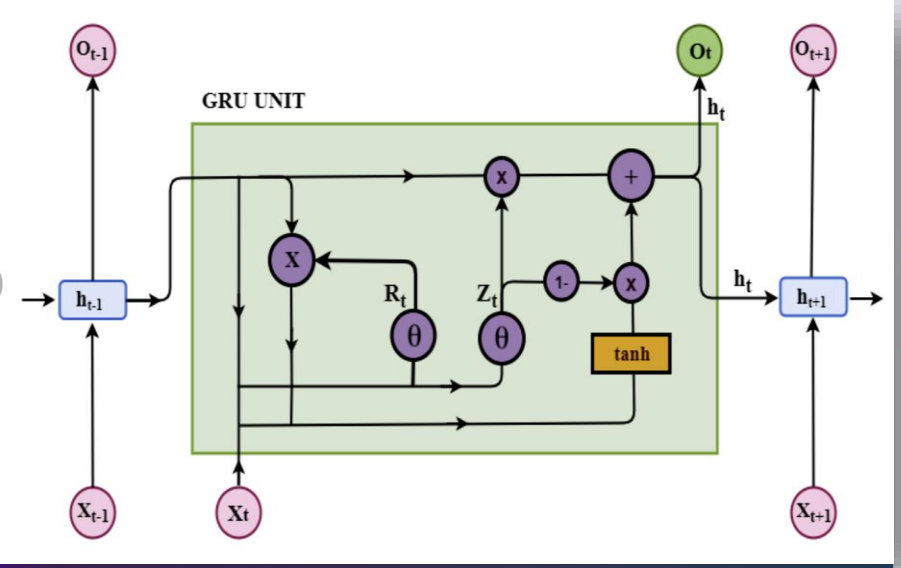
Architecture

EfficientNet scales depth, width, and resolution uniformly via a compound coefficient φ, building on MobileNetV2's MBConv blocks and squeeze-and-excitation (340 layers) to balance accuracy and efficiency

Deepfake Detection Use

EfficientNet variants (e.g., B4) serve as lightweight yet highly accurate backbones in deepfake classifiers, outperforming many earlier CNNs on FF++ and Celeb-DF benchmarks when fine-tuned for fake-real discrimination

GRUS



GRUs (Gated recurrent units)

GRUS are a type of RNN (Recurrent Neural Network) that models temporal dependencies in sequences, like video frames.

How they help:

- Deepfakes can have temporal inconsistencies—like unnatural blinking, misaligned lips, or inconsistent head movements.
- GRUs capture motion dynamics and continuity across frames, which can help detect such issues.

Architecture Summary:

- Frame-wise features (often extracted by CNNs or ViTs) are fed as a sequence into GRUs.
- GRUs learn temporal patterns and output a final classification vector or time-dependent predictions.

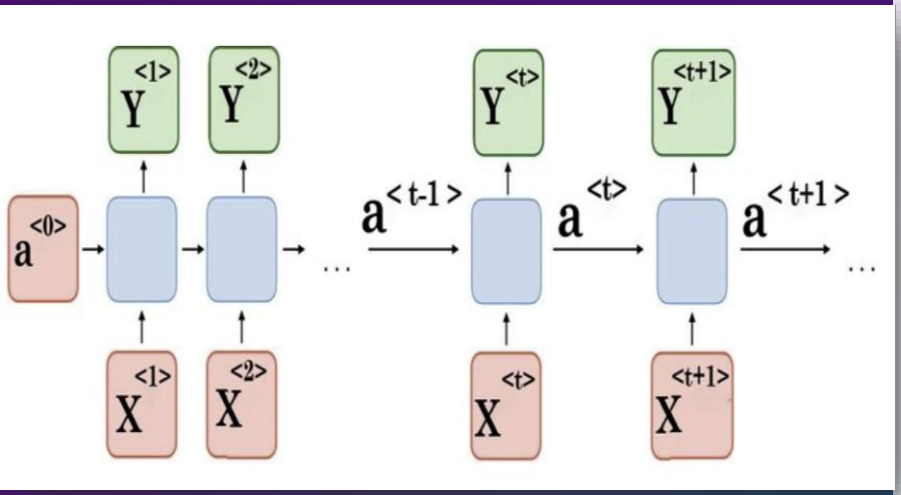
Use case in our ensemble:

The GRU models process sequences of frame-level features and predict whether the video segment is real or fake.

Deepfake Detection Use

GRU layers, often paired with CNN feature extractors (e.g., InceptionV3), process sequences of frame embeddings to detect temporal artefacts in video, improving classification speed and accuracy

LSTM



LSTM (Long Short-Term Memory):

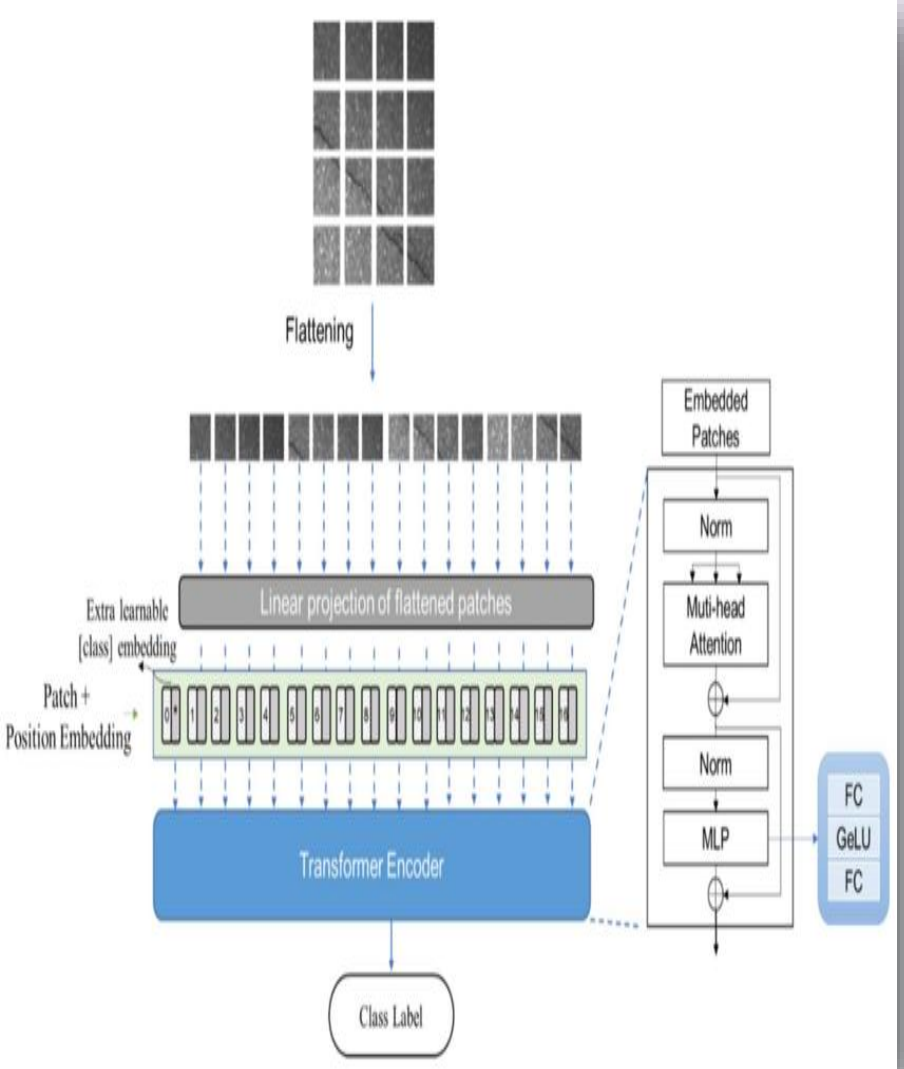
Architecture

An LSTM unit maintains a cell state and uses three gates—input, forget, and output—to regulate information flow, effectively mitigating vanishing/exploding gradients in sequence modeling

Deepfake Detection Use

In deepfake pipelines, LSTMs intake frame-level feature vectors (often from a CNN backbone) and learn temporal dependencies—like unnatural transitions in facial expressions—to classify video segments as real or fake

VISION TRANSFORMERS



Architecture

ViTs split an image into fixed-size patches, linearly embed them, add positional encodings, and feed the resulting sequence into standard transformer encoder blocks (multi-head self-attention + feedforward layers).ViTs treat an image as a sequence of patches (like tokens in NLP) and use self-attention mechanisms to model relationships between them.

How they help:

- Deepfakes often have visual inconsistencies like unnatural textures, blurring, or artifacts.
- ViTs are excellent at capturing global patterns and relationships, making them ideal for spotting such artifacts in individual frames.

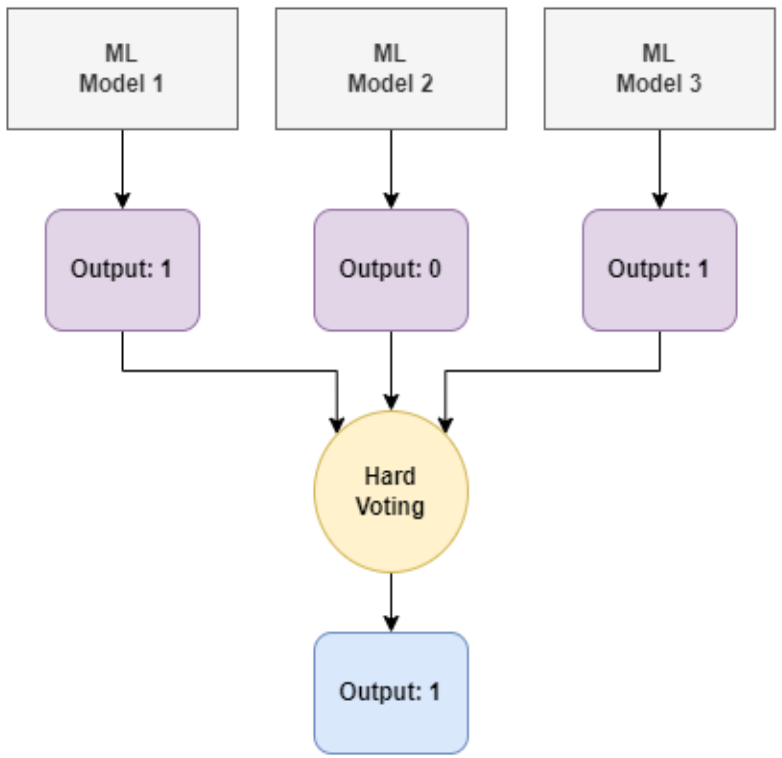
Architecture Summary:

- Input image is divided into fixed-size patches.
- Each patch is embedded and positional encodings are added.
- Transformer encoder layers apply multi-head **self-attention and feedforward layers.**
- The output is passed through a classification head.

Use case in ensemble:

Both ViTs analyze frames independently and output a probability of the frame being real/fake. These probabilities are passed to the ensemble.

ENSEMBLE



Hard Voting (Majority Voting)

- Each model casts a “vote” for real vs. fake.
- The final class is the one with the most votes.
- Why: Extremely simple, reduces the effect of any one model’s misclassification.

Soft Voting (Average Probabilities)

- Average the predicted probabilities (e.g. “fake” score) from each model, then threshold.
- Why: Preserves confidence information; often outperforms hard voting when models are well-calibrated.

Weighted Voting / Weighted Averaging

- Assign each model a weight based on its validation accuracy (or AUC), then do voting or probability averaging.
- Why: Gives more influence on stronger models, pulling the ensemble toward their decisions.

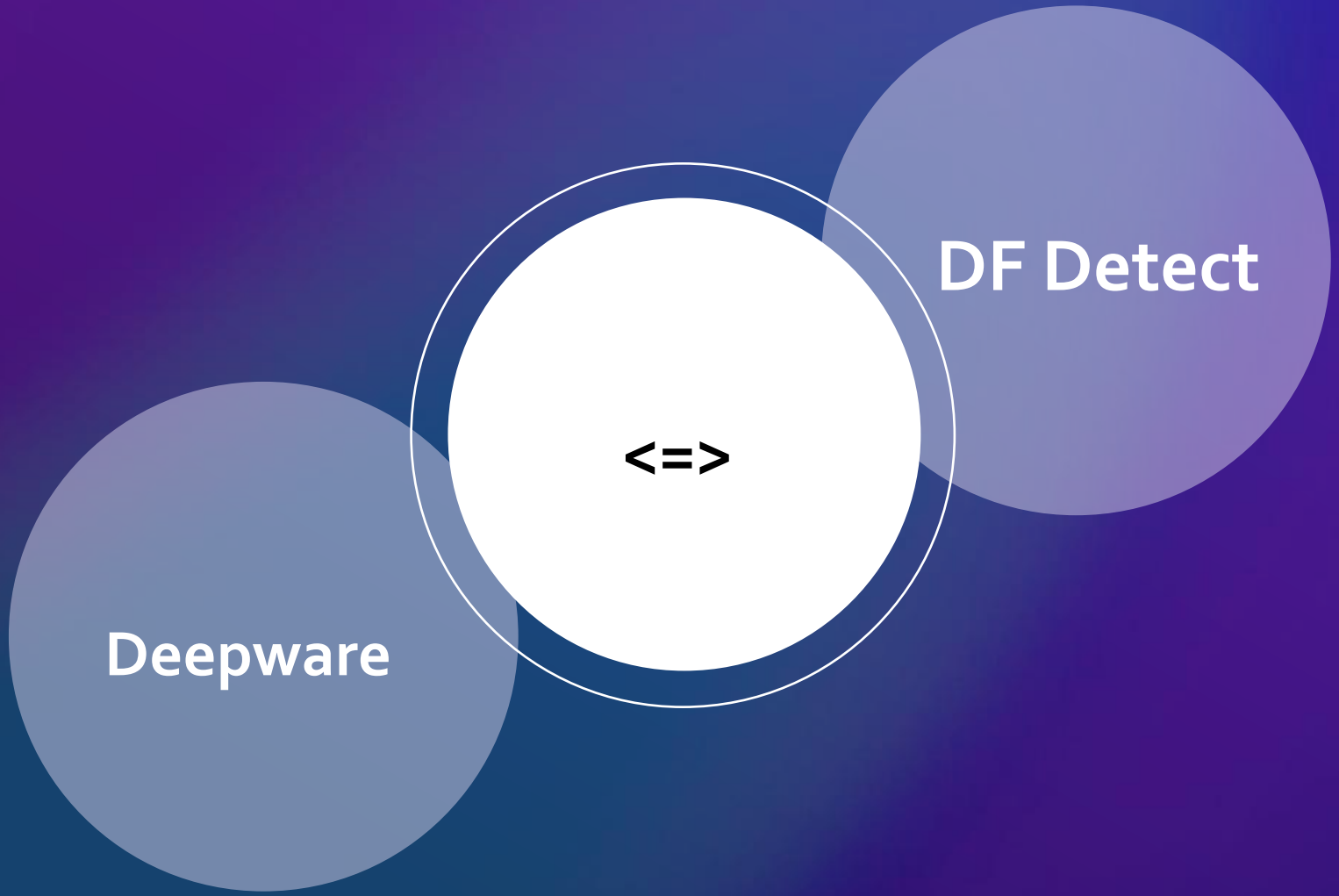
COMPARISON WITH EXISTING TOOLS

- **DF Detect**

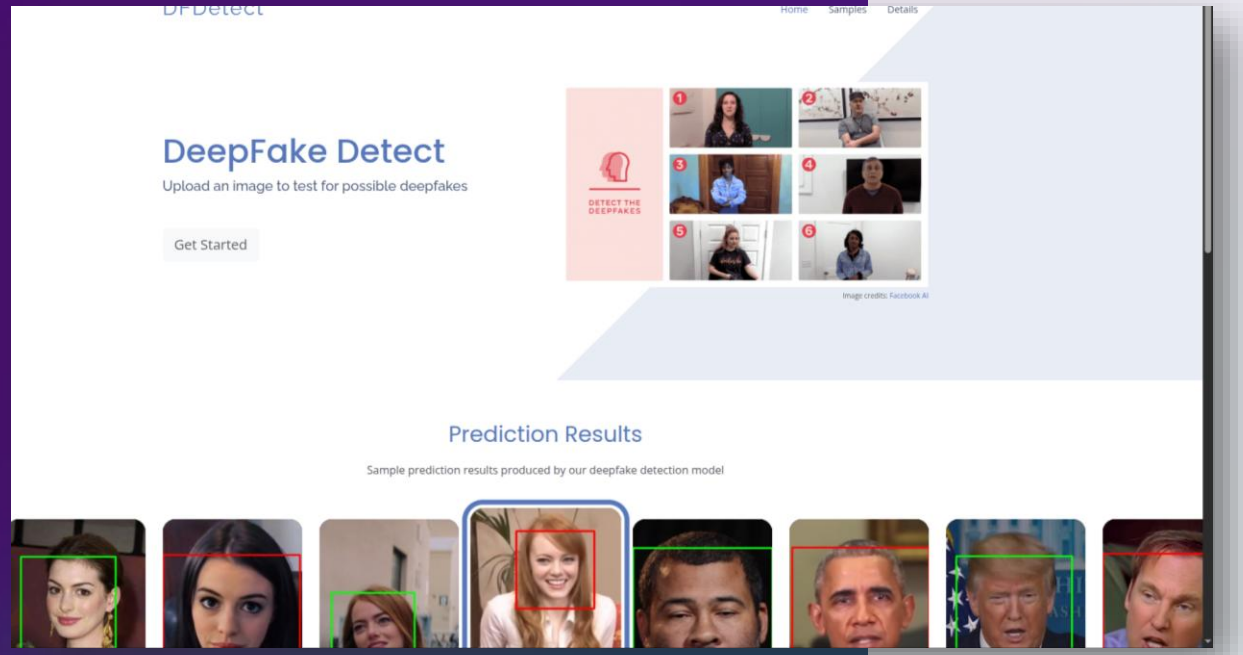
  Test images for possible deepfakes

- **Deepware**

  Scan & Detect Deepfake Videos



DF Detect



Tool:
https://deepfake-detect.com/

Key Features:

- A deepfake detection model based on EfficientNet, optimized for high accuracy.
- Designed to distinguish real vs. fake (GAN-generated/StyleGAN) face images.
- Achieves state-of-the-art performance compared to older models like XceptionNet and MesoNet.

Experimental Results:

- Input Size: 128×128 pixels
- Batch Size: 32
- Optimizer: Adam (LR = 0.0001)
- Dropout Rate: 0.5
- Regularization: L2 (λ = 0.001)

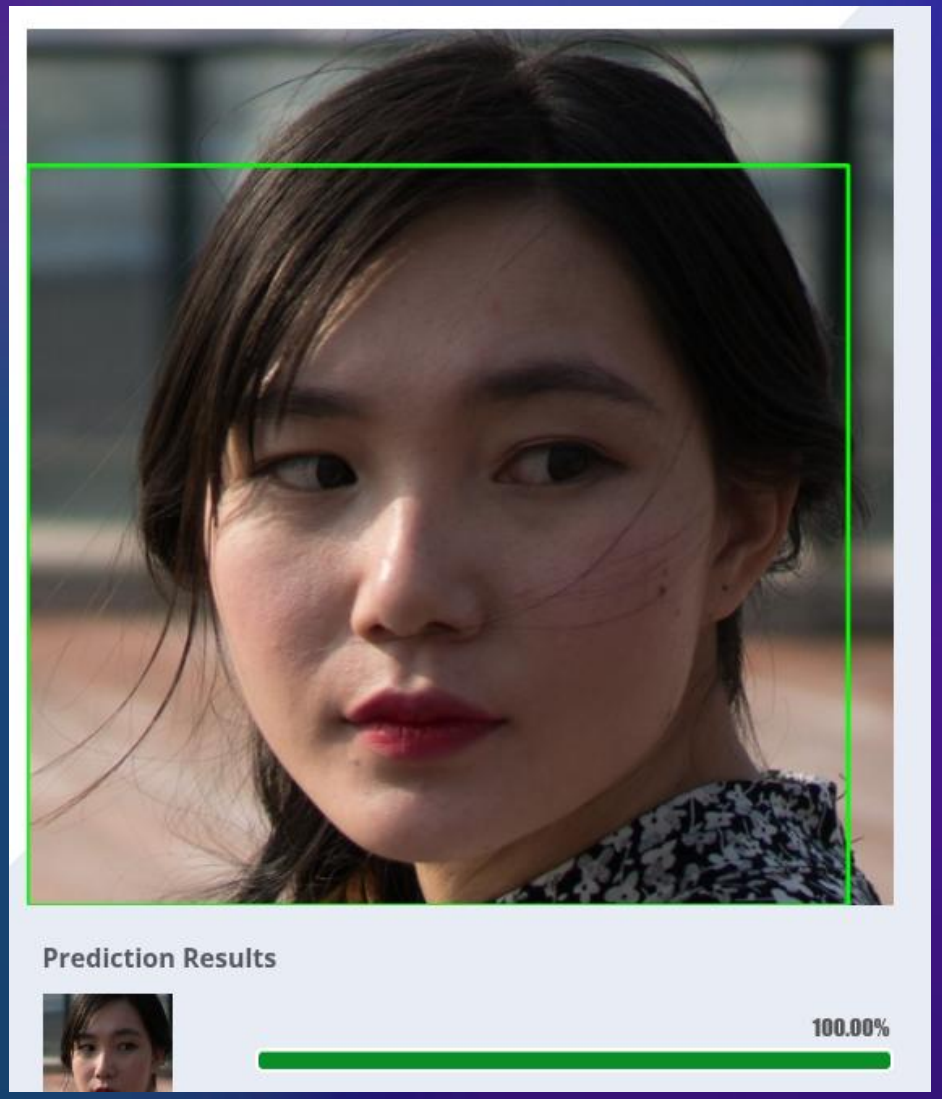
Performance Metrics:

- Accuracy: 96.36%
- Precision: 94.95%
- Recall: 97.94%

Developed Model

```
Votes: ['Real', 'Real', 'Real']
🏆 Final Decision: Real
real_31.jpg → Predicted: Real | Actual: Real
🖼️ Detected: Image Input
```

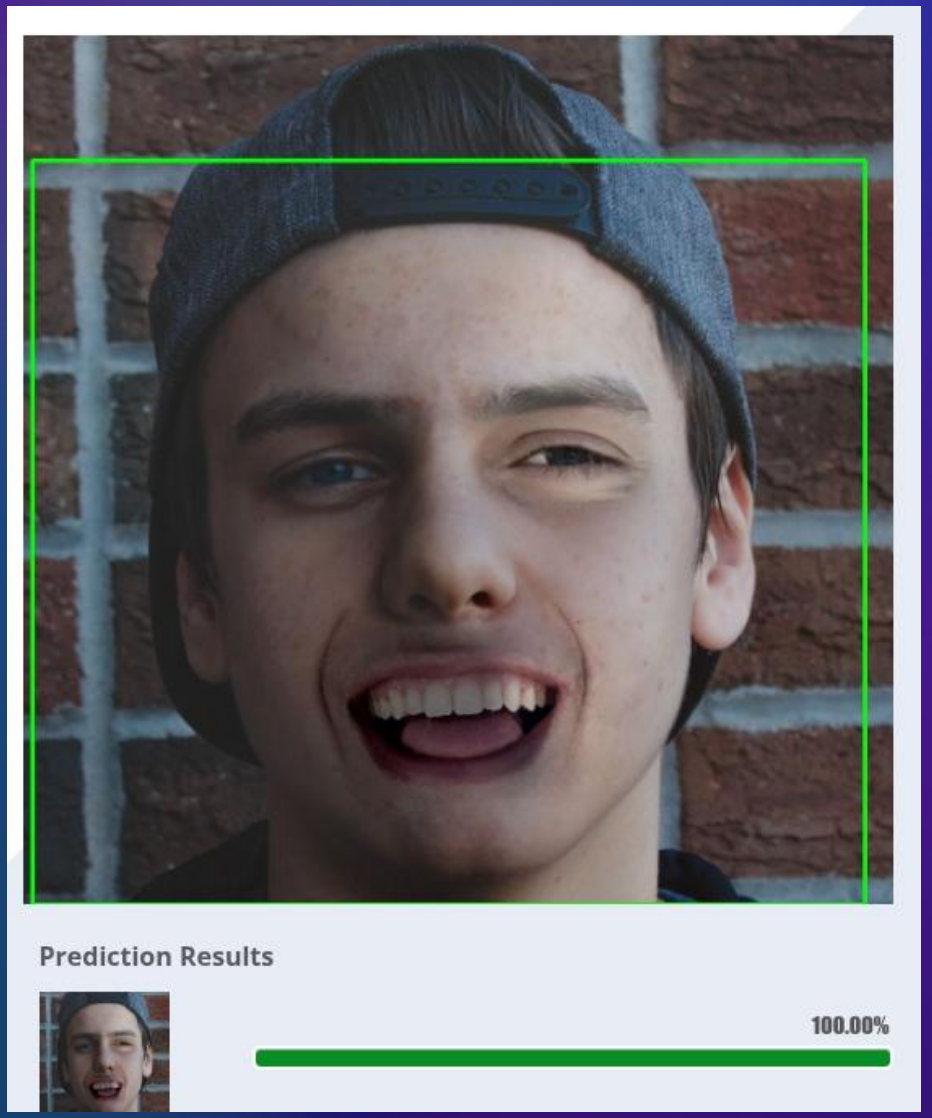
DF Detect



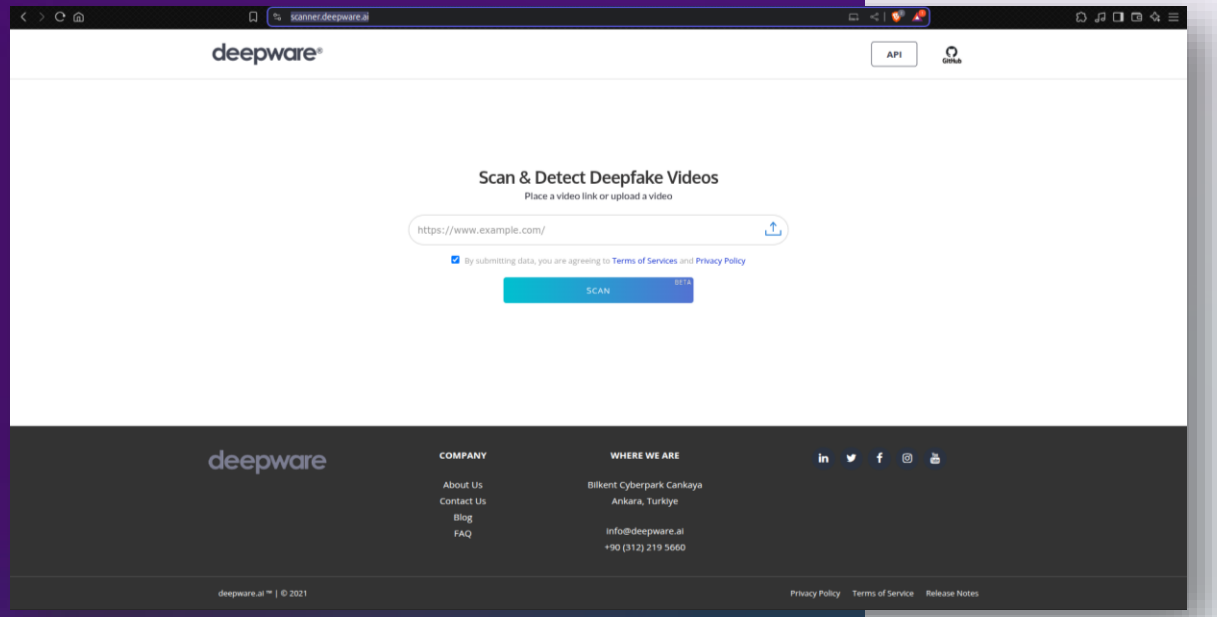
Developed Model

```
Votes: ['Fake', 'Fake', 'Real']
🏆 Final Decision: Fake
fake_73.jpg → Predicted: Fake | Actual: Fake
🖼️ Detected: Image Input
```

DF Detect



Deepware



Tool:
https://scanner.deepware.ai/

Overview:

- Free, AI-powered tool designed to detect deepfake videos by analyzing facial manipulations.
- Developed by Turkish cybersecurity company Deepware.
- Targets synthetic media threats in videos from platforms like YouTube, Facebook, and Twitter.

Key Features

- AI-Powered Deepfake Detection – Identifies face-swapped and AI-manipulated videos.
- Free & Web-Based – No installation needed; upload videos or paste URLs for scanning.
- Real-Time Analysis – Scans videos at 1 FPS for quick results.
- Enterprise Integration – Offers API/SDK for businesses and researchers.

Limitations:

- Max 10-minute video length.
- Works best on high-resolution (≥1080p) videos.
- Probabilistic Results – Provides confidence scores (e.g., "50% likely fake").

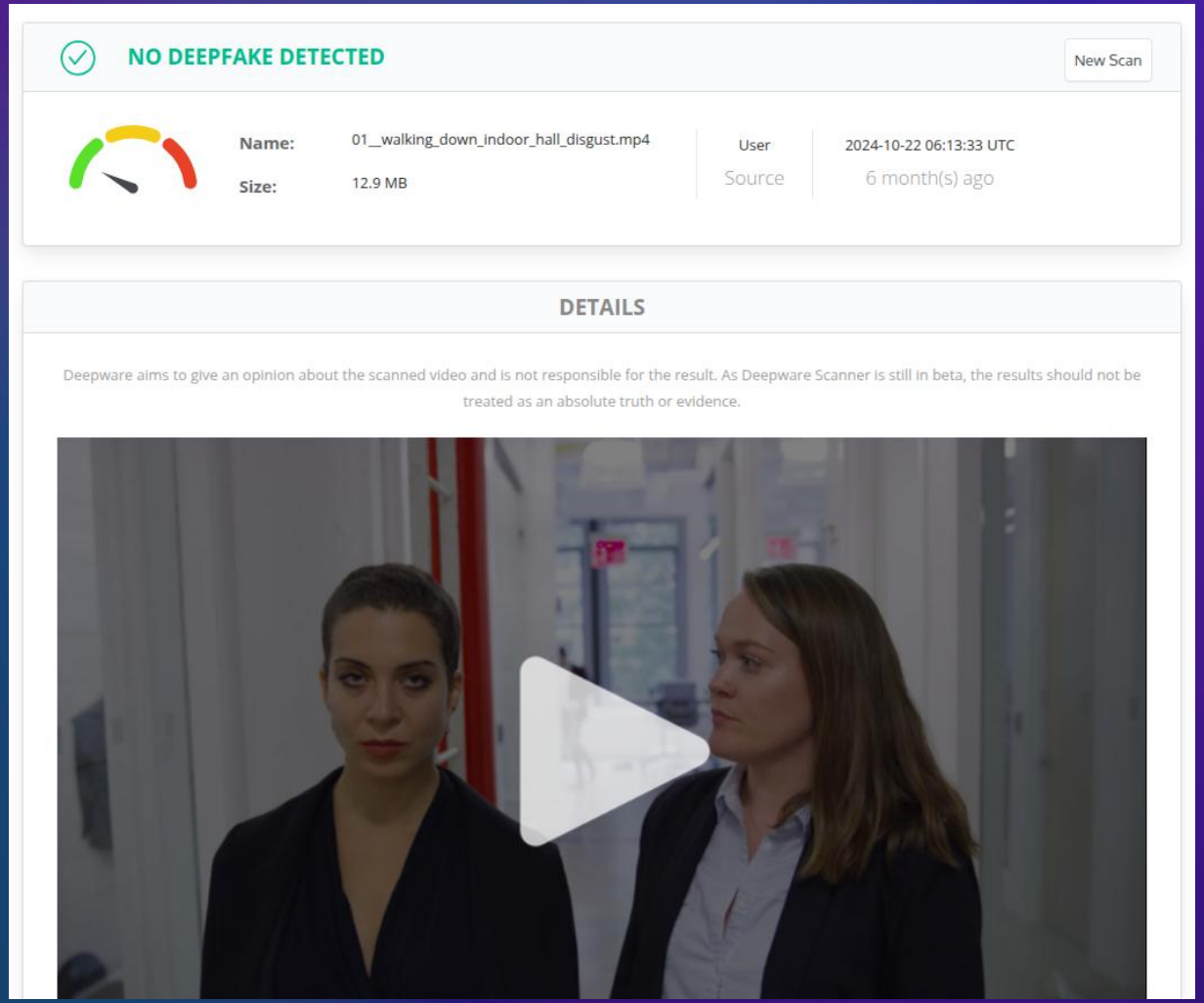
Developed Model

Choose Files sample_video_real.mp4

- **sample_video_real.mp4**(video/mp4) - 13523031 bytes, last modified: 2/13/2024 - 100% done

```
Saving sample_video_real.mp4 to sample_video_real.mp4

Analyzing: sample_video_real.mp4  (video)

🗳️  Votes → Code1: Real | Code2: Real | Code3: Real
🏆 Final Decision: Real
```

Deepware



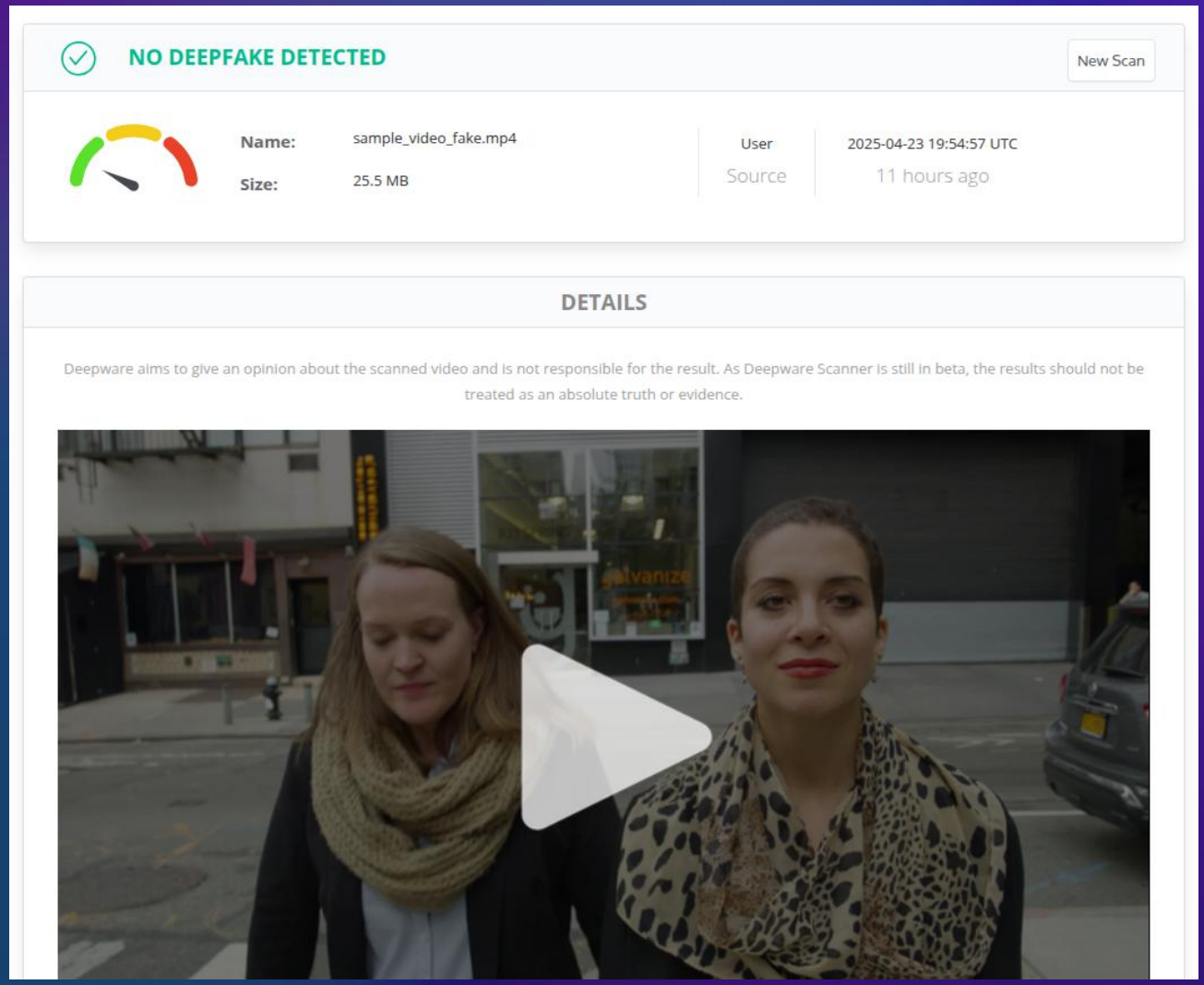
Developed Model

Choose Files sample_video_fake.mp4

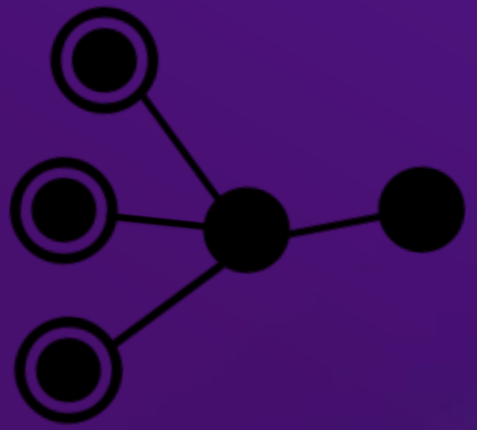
- **sample_video_fake.mp4**(video/mp4) - 26712461 bytes, last modified: 2/13/2024 - 100% done

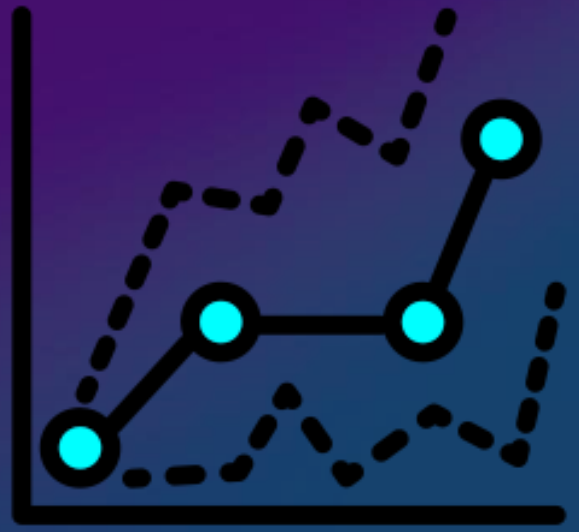
```
Saving sample_video_fake.mp4 to sample_video_fake.mp4

Analyzing: sample_video_fake.mp4  (video)

🗳️  Votes → Code1: Real | Code2: Fake | Code3: Real
🏆 Final Decision: Fake
```

Deepware



PREDICTION RESULTS





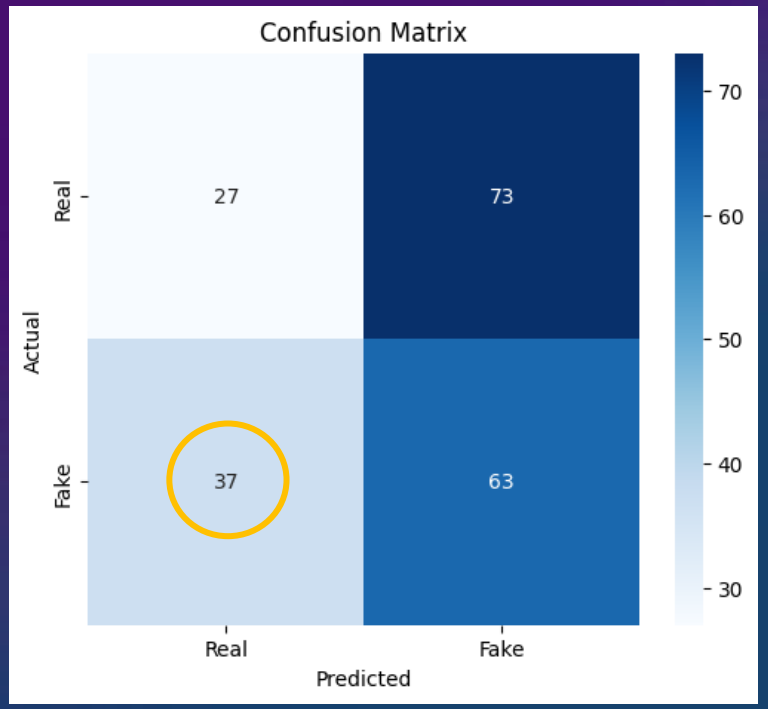
Weights

ViT	XceptionNet (on .mp4 files)	XceptionNet (on sequential images)	Accuracy
2	3	0	53.5
1	2	0	53.5
0	1	0	53.5
2	2	0	49.5
1	1	0	49.5
3	2	1	48.5
1	0	1	47
2	2	1	46.5
1	1	1	46.5
0	1	1	46
6	1	8	45
5	3	1	45
1	0	0	45
0	0	1	45

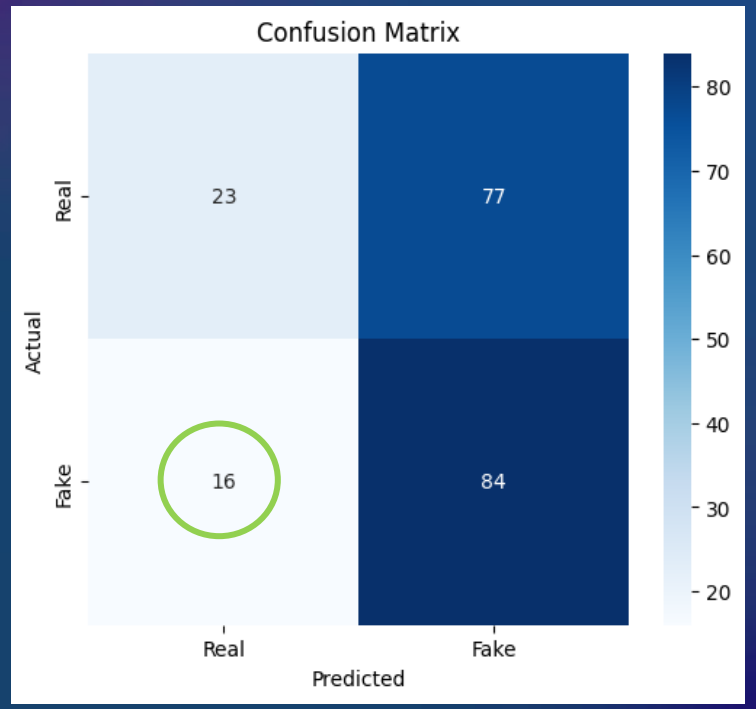
Rejecting Model 3 & Assigning higher weight to Model 2

ViT	XceptionNet (on .mp4 files)	XceptionNet (on sequential images)	Accuracy
40% 2	60% 3	0	53.5
33.33% 1	66.67% 2	0	53.5
0	1	0	53.5

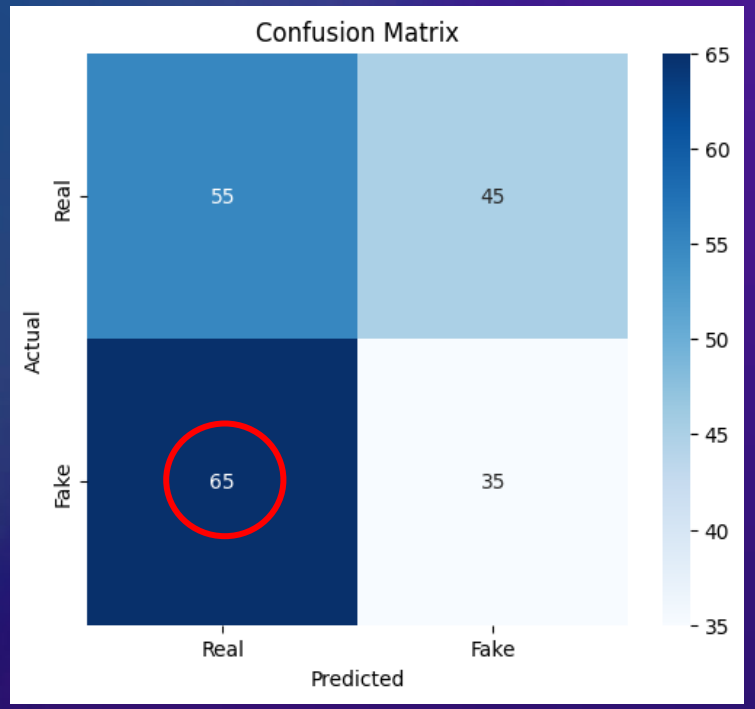
1 0 0 (recall=63%)



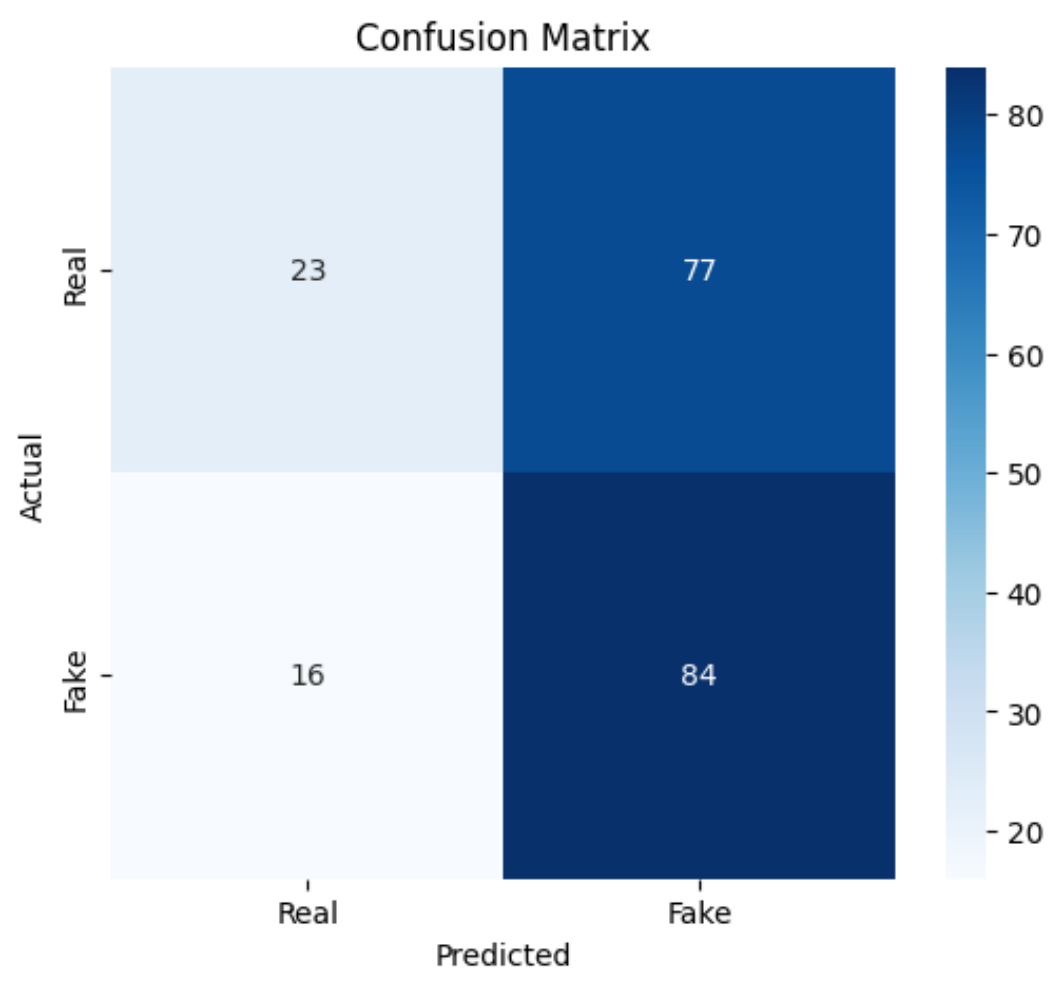
0 1 0 (recall=84%)



0 0 1 (recall=35%)



120

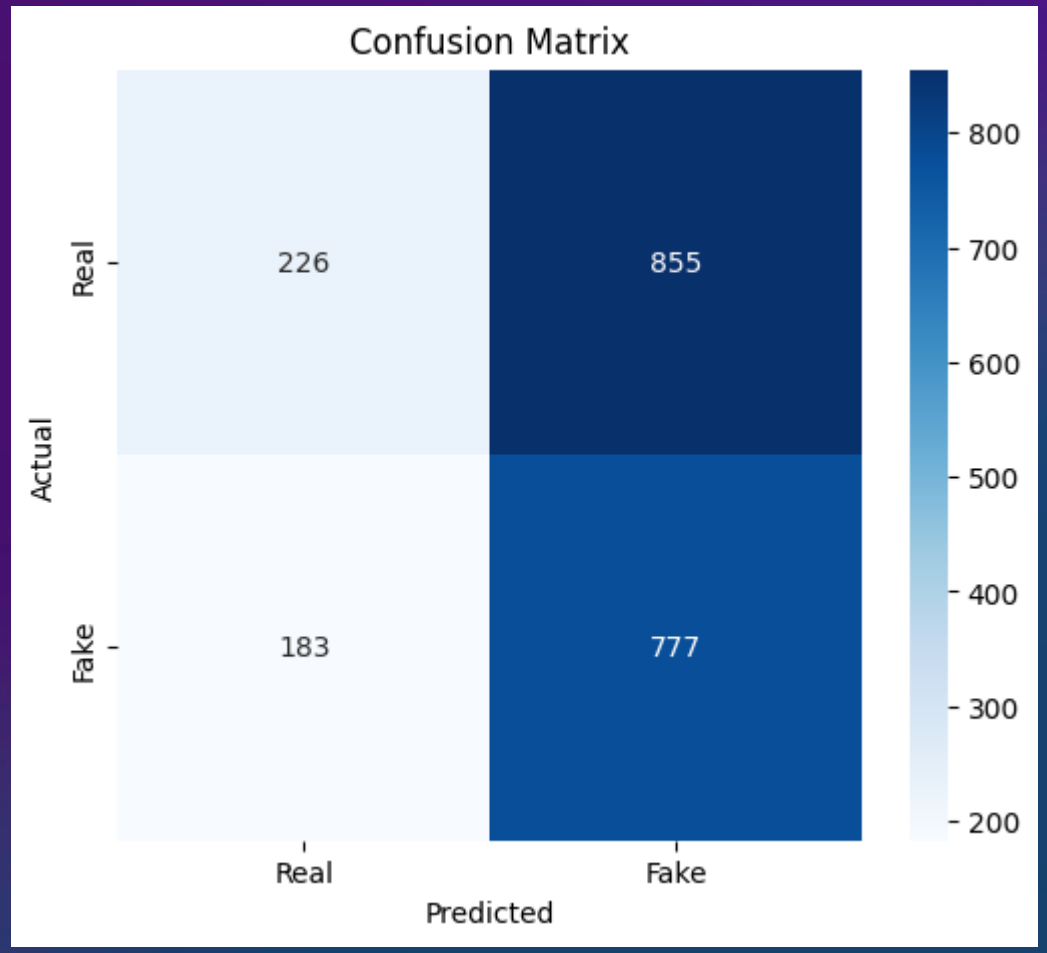


Tested on 200 images:

Recall = 84 / (84+16)

=84%

120



Tested on 2041 images:

Recall = 777 / (777+183)

=80.93%

Future Scope for Improvement

Adversarial & Robustness-Driven Training

- **Adversarial Augmentation:** Generate “hard” examples via adversarial attacks on your detector and include them in training.
- **Generative Augmentation:** Train a small generator to produce “challenging” fakes specifically designed to fool your ensemble.
- Anticipating and hardening against the next wave of deepfake techniques will make your model far more resilient in the wild.

Multi-Modal Fusion (Audio + Video)

Incorporate an audio-branch—e.g. a lightweight wav2vec or Audio Spectrogram Transformer—alongside your visual pipeline, then fuse predictions at frame, clip, or embedding level.
Many deepfakes only manipulate the face/video, leaving subtle audio artifacts (lip-sync errors, inconsistent background noise). A true multi-modal model is far more robust against attackers who focus on one modality.

Explainability & Uncertainty Estimation

- Embed attention-map visualization (Grad-CAM, attention rollout) to highlight suspicious regions.
- Incorporate Bayesian layers or Monte Carlo dropout for per-sample uncertainty scores.
- In many real-world applications (journalism, forensics), being able to explain why a clip was flagged—and how confident you are—can be just as crucial as raw accuracy.

THANK YOU