Real or Fake? An In-Depth Exploration of Faces in the Era of Deepfakes

Distinguishing between real and fake images

BARRACHIN Carlyne

DATASET

'140k Real and Fake Faces'

Flickr

IMAGES & VIDEOS HOSTING

StyleGAN

- NEURAL NETWORK
- ULTRA-REALISTIC

256 pixels

VARIATIONS
- AGE
- ETHNICITY
- BACKGROUND
- ACCESSORIES

3 FOLDERS : balance of classes ?

***Figure 1 - Distribution** of Real and Fake Images in each Dataset*

Figure 2 - Real** images from the **training dataset

Figure 3 - Fake** images from the **training dataset

Figure 4 - Real** images from the **test dataset

Figure 5 - Fake** images from the **test dataset

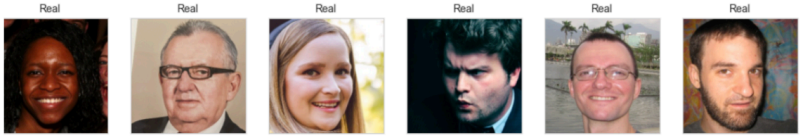

Figure 6 - Real** images from the **validation dataset

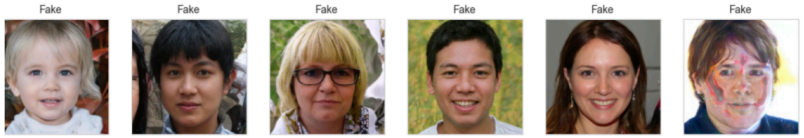

Figure 7 - Fake** images from the **validation dataset

Models

Keras | PyTorch

FEATURE	RESNET50	EFFICIENTNETB0	VIT (BASE, PATCH16, 224)
YEAR INTRODUCED	2015	2019	**2020**
ARCHITECTURE TYPE	CNN (Deep Residual Network)	CNN (Convolutional Neural Network)	Transformer (Vision Transformer)
NUMBER OF PARAMETERS	25.6M	**5.3M**	86M
INPUT IMAGE SIZE	224 x 224	224 x 224	224 x 224
FLOPS	~4.1B	**~0.39B**	~17.6B
STRENGTHS	Robust deep learning	Energy-efficient	Excellent performance on **large-scale** data
WEAKNESSES	Heavy, resource-intensive	Less effective on complex large datasets	Requires large datasets for good generalization
TRAINING COMPLEXITY	Medium	**Low**	High
KEY STRUCTURAL FEATURES	**Residual connections**	NAS-based architecture	**Patch** embedding and global self-attention
NUMBER OF LAYERS	50 layers	234 layers	12 Transformer encoder layers
LAYER TYPES	Convolutional layers, **residual blocks**	**MBConv** blocks with depthwise separable convolutions	Self-attention, **MLP** layers, **LayerNorm**
ATTENTION MECHANISM			**Multi-head** self-attention (12 heads)

COMPUTATIONAL DEMAND

Table 1 - Models Architecture comparison

Attention allows different **patches** of the image to be **linked together** to better understand the image as a whole, rather than treating each patch **separately** as in **CNNs**.

EXPERIMENTAL SETUP

Loss Function	BCE (Binary Cross-Entropy Loss)
Hyperparameters Optimization	ADAM
Multiple layers	Layers model Dimensionality Reduction Sigmoid Activation Function : single output between 0 and 1
Batch Size	Train and Validation: 32, Test: 1
Epoch	CNNs: 5, ViT: 3

Training and Validation

Loss

Accuracy

EfficientNetB0

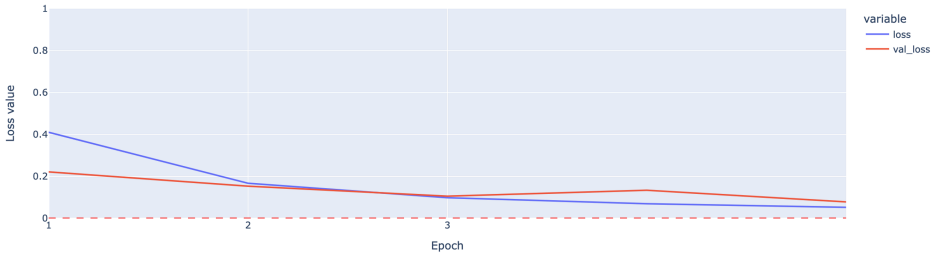

Figure 8** - Training and Validation **Loss** for **EfficientNetB0

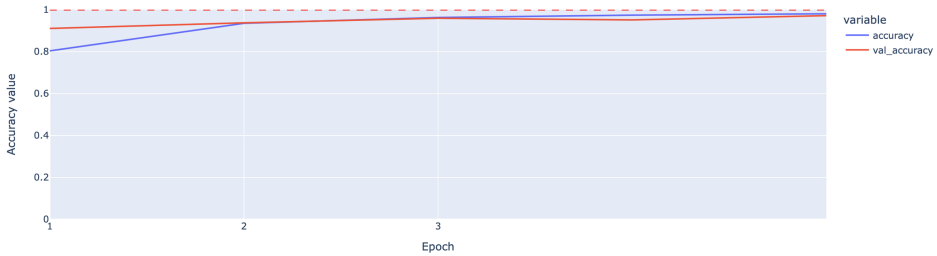

Figure 9** - Training and Validation **Accuracy** for **EfficientNetB0

ResNet50

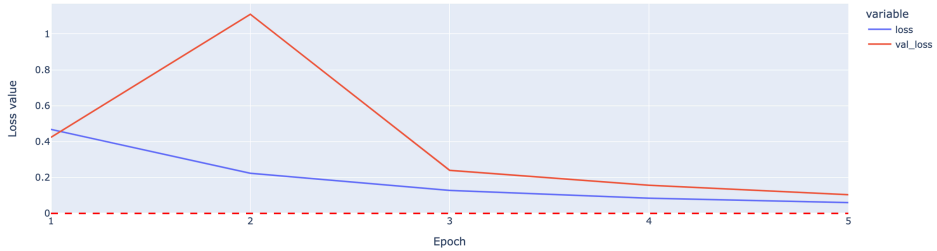

Figure 11** - Training and Validation **Loss** for **ResNet50

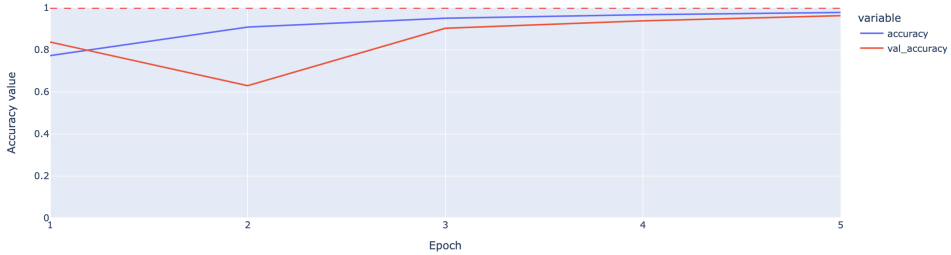

Figure 10** - Training and Validation **Accuracy** for **ResNet50

ViT

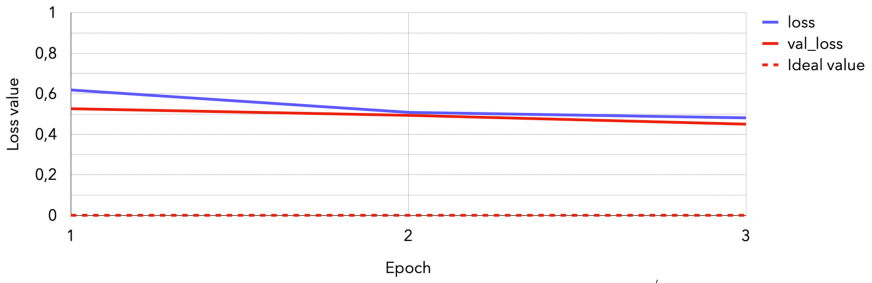

Figure 13** - Training and Validation **Loss** for **ViT

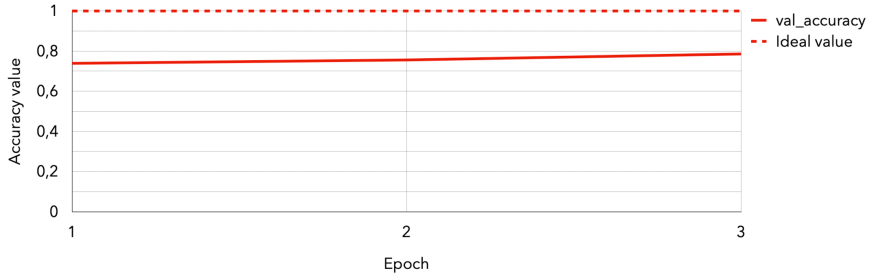

Figure 12** - Training and Validation **Accuracy** for **ViT

EfficientNetB0

97,17% **accuracy**

	precision	*recall*	*f1-score*
Fake	0.98	0.97	0.97
Real	0.97	0.98	0.97

Table 2 *- Precision, Recall and F1-Score EfficientNetB0*

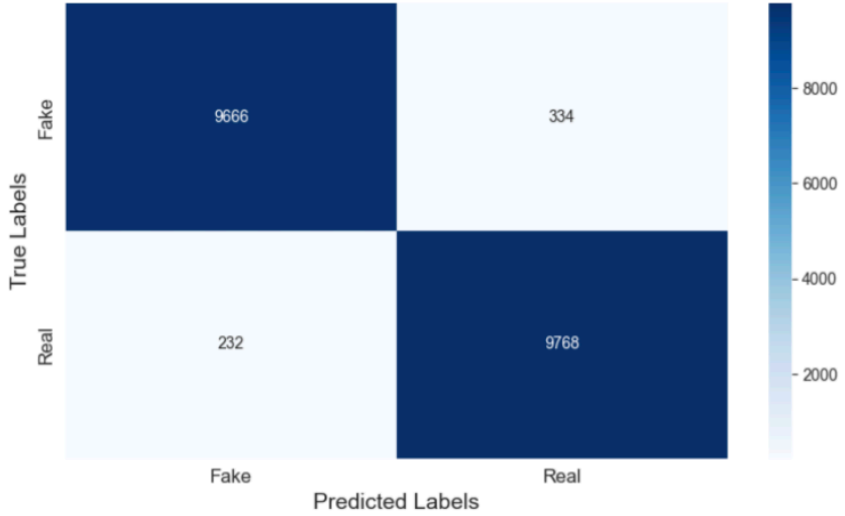

Figure 14 *- Confusion Matrix EfficientNetB0*

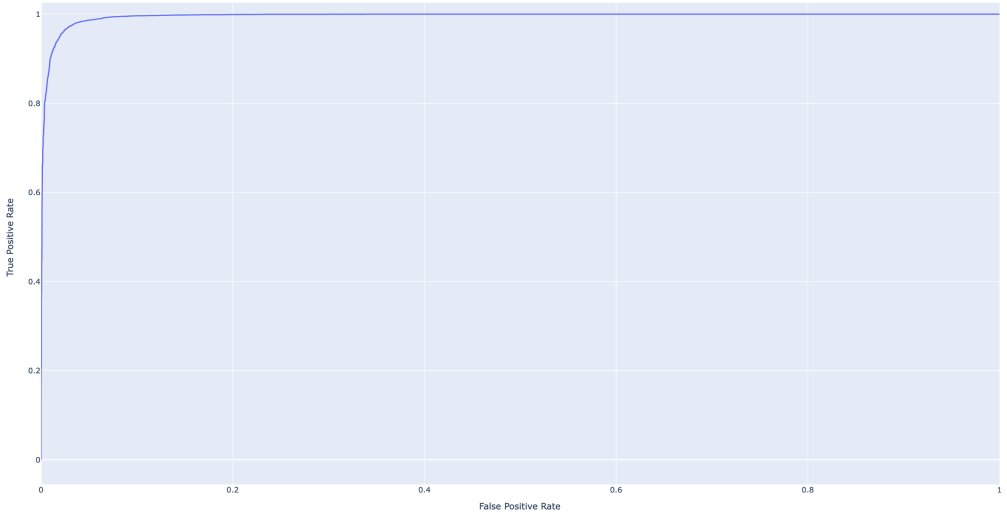

Figure 15 *- ROC-AUC Curve with* ***AUC = 0.9956***

Area Under the Curve: 1 indicates perfect class separation

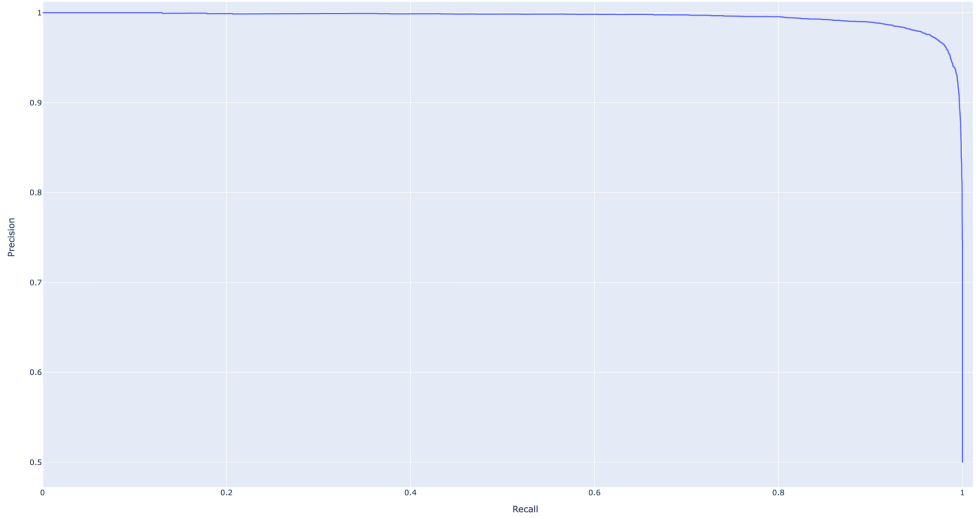

Figure 16 *– Precision-Recall Curve with* ***AP = 0.9951***

Average Precision

ResNet50

96,41% **accuracy**

	precision	*recall*	*f1-score*
Fake	0.98	0.95	0.96
Real	0.95	0.98	0.96

Table 3 *- Precision, Recall and F1-Score ResNet50*

Figure 17 *- Confusion Matrix ResNet50*

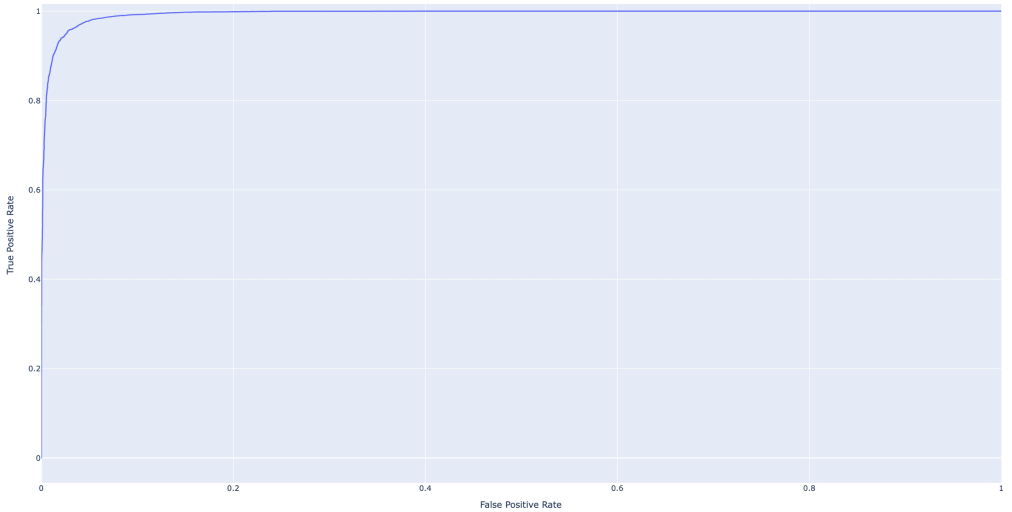

Figure 18 *- ROC Curve with* ***AUC = 0.9944***

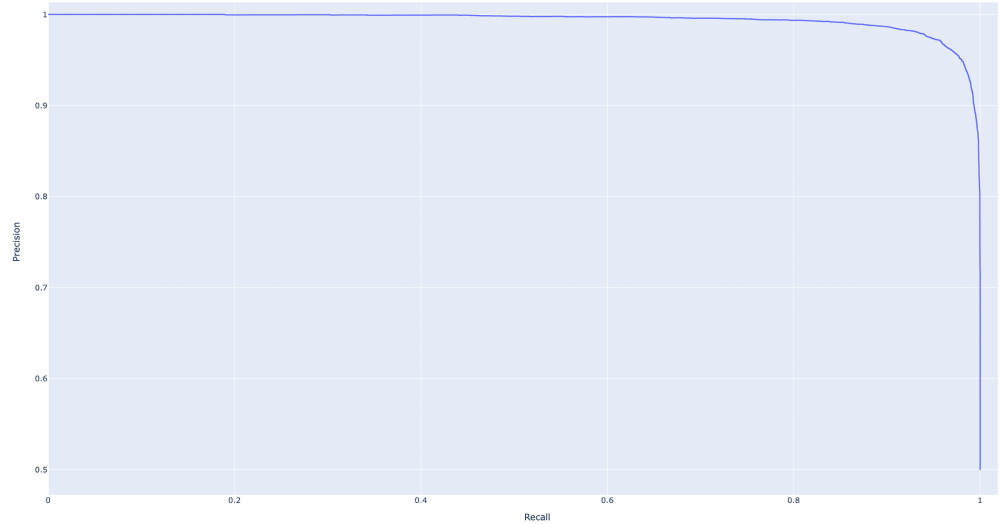

Figure 19 *- Precision-Recall Curve with* ***AP = 0.9939***

ViT

78,65% **accuracy**

	precision	*recall*	*f1-score*
Fake	0.86	0.68	0.76
Real	0.74	0.89	0.81

Table 4 *- Precision, Recall and F1-Score ViT*

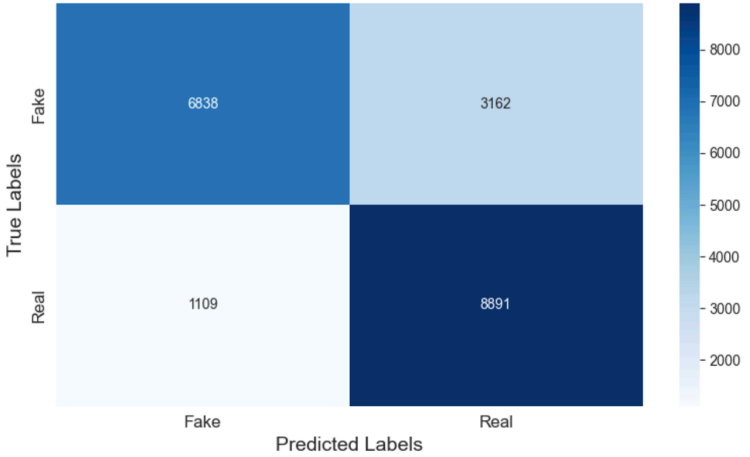

Figure 20 *- Confusion Matrix ViT*

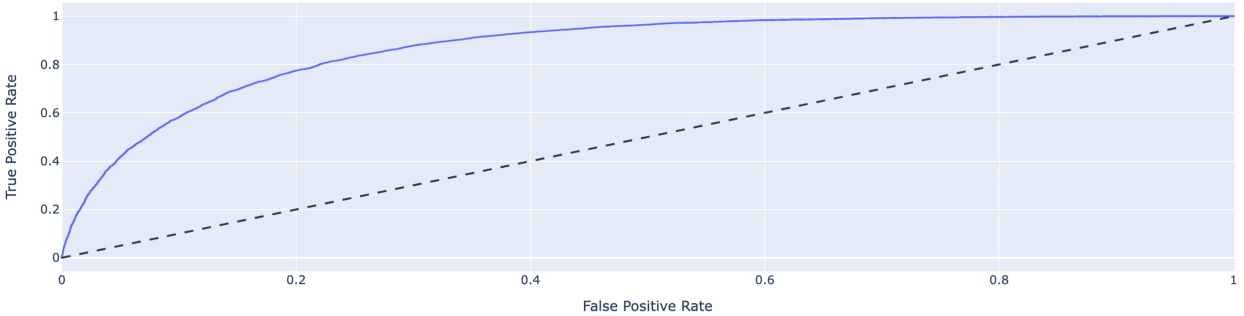

Figure 21 *- ROC Curve with* ***AUC = 0.8717***

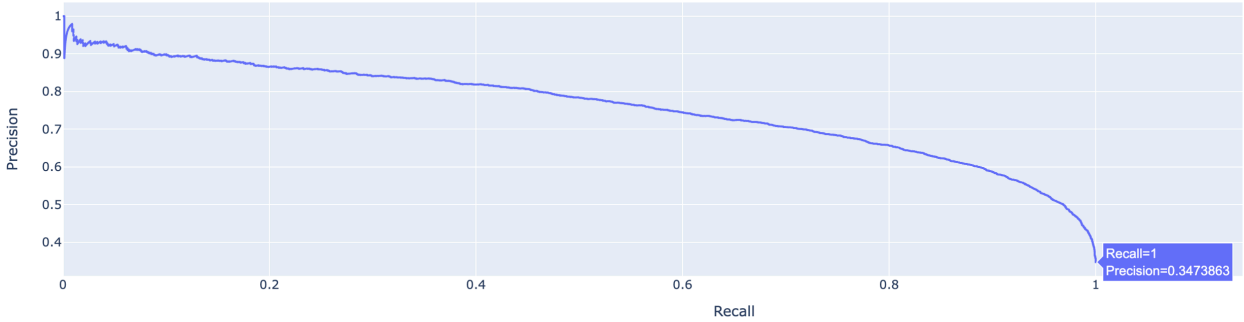

Figure 22 *- Precision-Recall Curve with* ***AP = 0.7611***

Errors

UNUSUAL HAIR COLOR

DARK LIGHTING
SHADOWS ON FACES
UNIFORM BACKGROUND

Figure 23 - *Misclassified Images* ***EfficientNet***

Figure 24 - *Misclassified Images* ***ResNet***

FACE ORIENTATION
slighty turned

BLURRED BACKGROUND

Figure 25 - *Misclassified Images* ***ViT***

Real-Life Experiment

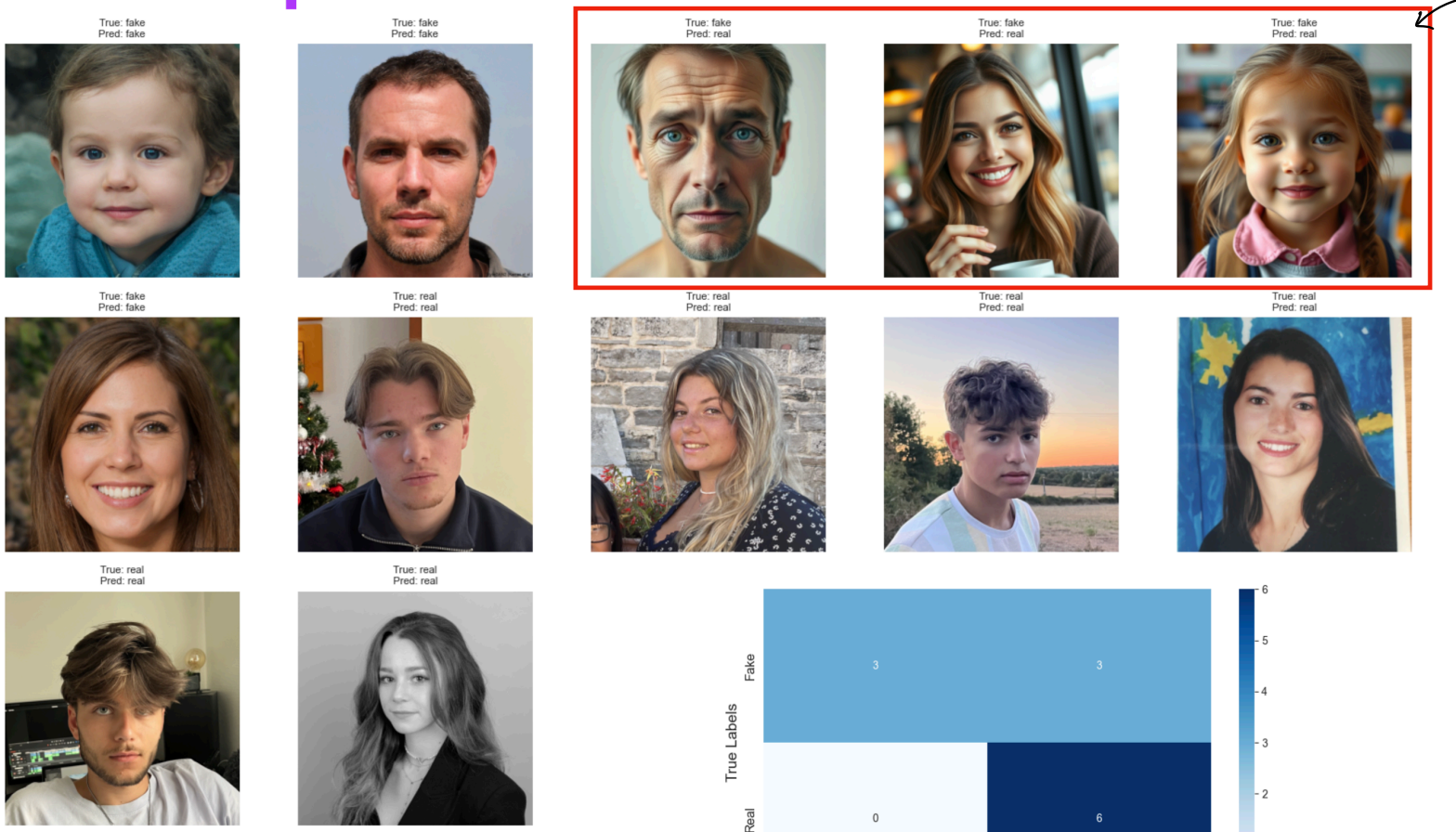

Figure 26 - *Confusion Matrix*

Comparison

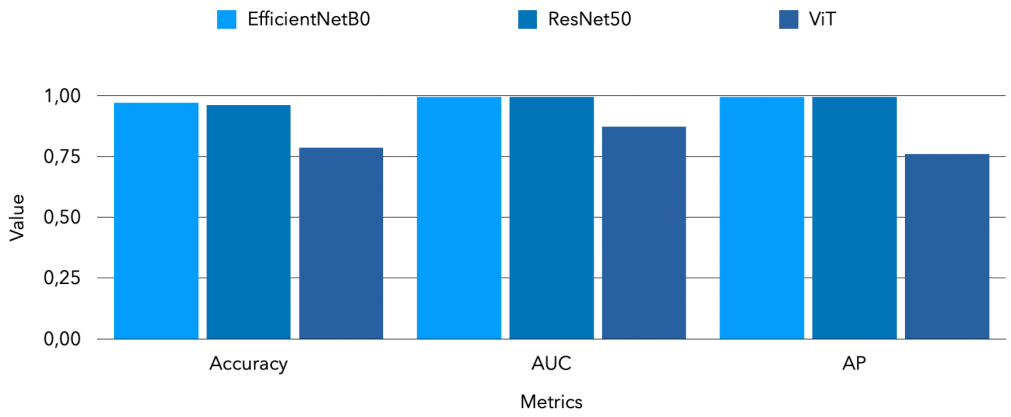

Figure 27 *– Accuracy, AUC and AP Comparison*

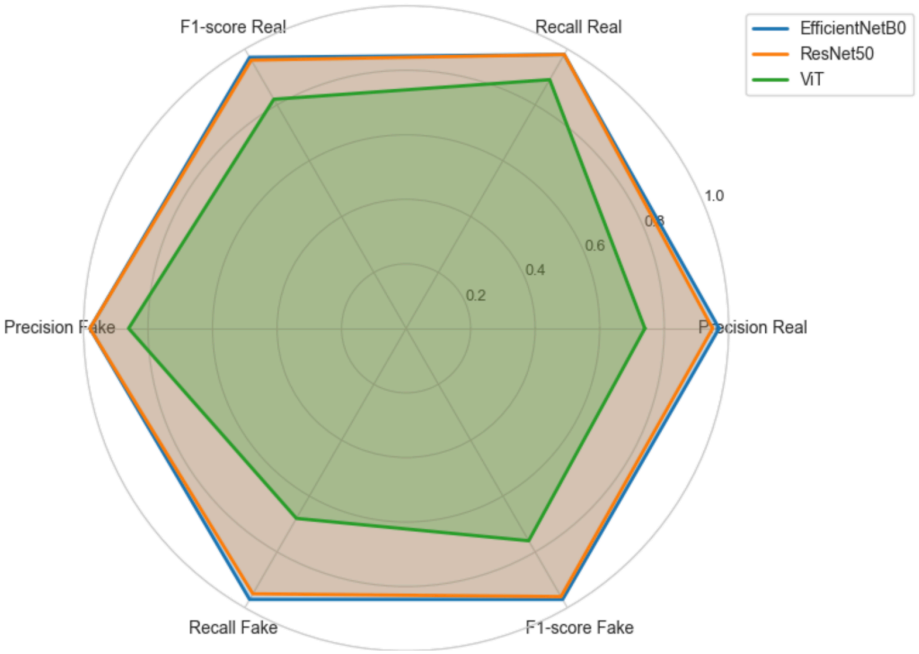

Figure 28 *– Radar plot of model performance*

EfficientNetB0: Best balance of efficiency and accuracy.

ViT: High resource demands, limited training.

Conclusion

Recents Models and Datasets

Hybrid Architectures:

Combines CNN (local features) + ViT (global relationships).

Applications:

Strengthen privacy, combat misinformation, enhance digital security.

Thank you !

BARRACHIN Carlyne - ERASMUS