Deepfake Face Detection: An Ensemble Framework for Generalized Classification in Biometric Verification Systems

by

Hilary Zen

S.B. Computer Science and Engineering
Massachusetts Institute of Technology, 2024

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2025

Authored by:	Hilary Zen Department of Electrical Engineering and Computer Science May 9, 2025
Certified by:	Amar Gupta Research Scientist, Thesis Supervisor
Accepted by:	Katrina LaCurts Chair Master of Engineering Thesis Committee

Deepfake Face Detection: An Ensemble Framework for Generalized Classification in Biometric Verification Systems

by

Hilary Zen

Submitted to the Department of Electrical Engineering and Computer Science
on May 9, 2025 in partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

ABSTRACT

Generation methods for deepfake images have advanced rapidly, and deepfake face images pose a critical security for biometric verification systems. Applications that rely on face recognition to grant access to sensitive data need to maintain high accuracy across a wide variety of deepfake generation methods, including novel and developing types that the application has not previously trained on. Current deepfake detection models achieve near-perfect accuracy on benchmark datasets, but do not perform as well on unseen types of deepfakes that were not part of their training dataset. We propose building an ensemble model with multiple base detectors, each trained on different generation model families to maintain high performance across many deepfake generation methods. Using four base models, including two models with the same architecture and training data, we exhaustively test all possible ensemble models. We find that combining similar base models trained on the same deepfake generation family does not improve performance compared to the individual base models. However, combining base models trained on different deepfake generation families leads to significant increases in accuracy and recall. Our ensemble framework provides a flexible and inexpensive solution in the ever-changing landscape of deepfake generation and security.

Thesis supervisor: Amar Gupta
Title: Research Scientist

Acknowledgments

Thank you to Dr. Amar Gupta, my thesis supervisor, for his support and guidance as I led a research project and wrote this dissertation. Throughout my masters, I gained invaluable experience in leading large groups, collaborating with researchers from different backgrounds, and writing academic papers. I could not have accomplished this without Dr. Gupta's advice and his trust in me.

I am incredibly grateful for the members and collaborators of my lab group. Thank you to Rachel Park for co-leading with me in the fall, brainstorming new directions and ideas, and leading work in IAP. Thank you to Rohan Wagh for joining me as a lead in the spring, and creating some of the data and evaluations that are a crucial foundation for this thesis. I also appreciate the contributions of Aiden Etheridge, Arashdeep Singh, Megan Sun, Jeffery Zhu, Abhitha Vegi, and Nancy Wang as members of my group throughout the year; and Dr. Rafael Palacios, Miguel Wanderley, Gustavo Bicalho, Lucas Carvalho, and Guilherme Rinaldo for their wonderful feedback and research ideas.

Finally, thank you to my family and friends for supporting me throughout my time at MIT. Our experiences have been the highlights of my last five years, and I could not have gotten to this point without all of you.

Contents

List of Figures

List of Tables

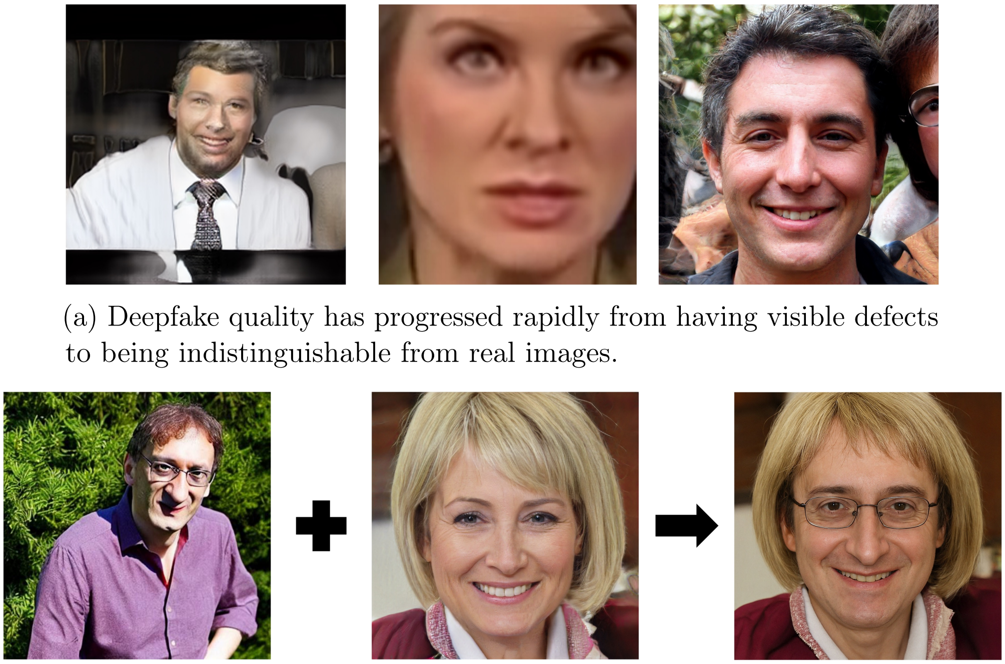
Chapter 1

Introduction

1.1 Deepfake Images

Deepfakes are images, video, and/or audio that have been edited or generated to represent people in places they were not actually at, or events that did not happen. The people in these deepfakes may be real people whose faces or voice have been edited into the deepfake, or they may be entirely computer-generated. Since this term was coined in 2017, the number of generation tools and quality of deepfaked media has grown exponentially. Deepfakes are now incredibly realistic and difficult to distinguish from unedited images.

This technology has opened up the ability to create completely false media that is still realistic and convincing, with huge political, social, and personal ramifications. Deepfake videos and audio have been used in state misinformation campaigns, including one that falsely claimed Ukraine's President Volodymyr Zelensky was asking citizens to surrender [1] and another that used generated characters by the AI company Synthesia [2]. As these campaigns become more widespread, they create loss of trust in the government and prominent news sources, as well as content on social media. Because there are very few legal regulations and protections in the United States, people who find themselves the subject of a deepfake have very little recourse in removing the media from the Internet or being compensated for the



(a) Deepfake quality has progressed rapidly from having visible defects to being indistinguishable from real images.

(b) In a faceswap image, the face from a source image (on far left) is placed into a target image (in the middle) to produce a deepfake (far right).

Figure 1.1: Six examples of deepfake face images

harm done to them. Finally, deepfakes are quickly becoming a common method for malicious actors to defraud individuals for money, critical information, or access to a protected system. In one extreme case, a worker was convinced to make a $25 million payment after a video call with several deepfake videos of colleagues [3]. Deepfake images and videos are also used by malicious actors in their attempts to get past facial recognition systems that protect account access and personal data, as we will discuss in Section 1.2.

1.2 Motivation

We focus on deepfakes within biometric verification systems and how deepfake images specifically can be used in attacks to gain access to sensitive accounts and information. Biometric verification systems use a person's unique biological features, like their facial structure or fingerprint, to identify them. The most common application is facial recognition to unlock a smartphone or account, but digital identity verification is part of many other systems and applications.

Particular industries that may be targeted by deepfakes include healthcare and government. These systems provide essential services, require a high degree of public trust, and store a lot of sensitive data. There is a high risk of malicious actors impersonating doctors or government officials to spread misinformation and cripple operations. Patients and citizens must be protected against having their data or private medical info stolen. Financial systems can also be targeted due to common usage of biometric verification and larger possible monetary gains. In all the aforementioned industries, it is crucial to minimize the losses that victims of deepfake identity fraud may experience. Biometric verification systems have a strong need to detect deepfakes from a wide array of tools. However, despite significant research in deepfake detection over the last few years, ensuring that models generalize to deepfake types outside their training data remains a difficult problem. Model development and retraining is also an expensive process, and may not be feasible with how quickly the field of deepfake generation advances. We aim to apply ensemble models in deepfake detection to build a model that can be adapted easily to new deepfake generation techniques while maintaining high performance.

1.3 Contributions

In this thesis we:

- categorize the main deepfake generation methods that are most popular today, and show that existing detection models do not perform well across all main deepfake types
- introduce a new methodology for selecting and evaluating deepfake detector models on a curated training dataset
- demonstrate that an ensemble of individual detectors significantly improves performance across a broad dataset, and even low-performing models can fill a generalization gap in a higher-performing model

Chapter 2

Related Work

This thesis focuses only on deepfake images, even though deepfake video and audio are becoming much more commonplace and also pose a security risk for biometric verification systems. In the next few sections, we review related research in the generation and detection of deepfake images. We also present the generalization problem and how ensemble models have been applied in related fields.

2.1 Deepfake Image Generation

We categorize the current deepfake generation methods into three families: GANs, VAEs, and diffusion.

2.1.1 GANs

Generative adversarial networks (GANs) use a generator and discriminator that compete against each other, with the aim of improving the generator model until it produces fake samples that the discriminator cannot distinguish from real samples [4]. One of the most influential developments in using GANs for deepfake generation was StyleGAN, which adapted the traditional generator architecture to add a style transfer algorithm before each convolution

[5]. Generators are often changing and improving to produce different artifacts [6].

2.1.2 Autoencoders and VAEs

Variational autoencoders (VAEs) expand on the autoencoder architecture to gain more powerful generative capabilities. They consist of an encoder-decoder pair where the encoder takes an image and outputs a distribution over a latent space, and the decoder attempts to recreate the original image from the latent distribution [7]. Many popular faceswapping tools utilize two encoder-decoder pairs, one pair trained on a source image and the second pair trained on a target image, to place the face of a source into a target image [8].

2.1.3 Diffusion

Diffusion models learn to denoise images, made noisy by a forward process, step by step in their reverse process to gradually reaching a final sample without noise [9]. They can be improved by incorporating classifier gradients into diffusion sampling, to provide information on what class an image is in [10]. The RePaint model demonstrates high generative capabilities given a masked image, even when a large majority of the image is noisy [11].

2.2 Detection Models

There are a large variety of model architectures used in deepfake detection, and several ways to categorize them. A 2022 survey [8] separated categories by what deficiency the model uses, with the most relevant categories for deepfake image detection being inconsistencies, unusual environment factors like lighting, GAN fingerprints or artifacts, and CNN-based models.

Inconsistencies in the face or background of the image are often used to detect faceswap images, where portions of a source and target image have been put together [12]. Patch-based models analyze an image in small regions to discover artifacts between the patches [13]. Models often use vision transformers (ViTs) which split an image up and use multi-head self-attention to collect local information from small regions [14, 15].

Wang et al. [16] performed a frequency analysis on multiple GANs and deepfake generation methods, finding dot or line patterns in many average image spectra from GANs. Other studies have explored GAN fingerprints are generated and how detection models can stay ahead of GAN improvements that aim to remove these fingerprints [17].

Many deep learning models have been developed for deepfake detection, with many models building on top of general image classification models like ResNet-50 [16]. Other architectures include a reduced DenseNet model with two input streams to take a fake and real image input together, making use of pairwise learning to learn common fake features [18]. As the field of deepfake detection and the size of models grows, unlabeled data may become more commonplace. Fung et al. [19] makes use of contrastive learning to teach the model how to distinguish real and fake images without labels. Comparatively few detectors have looked at unsupervised learning, but this area may rapidly grow in the near future.

2.3 Generalization and Ensemble Methods

Model generalization is a known challenge in deepfake detection and has significant motivation and existing research. As outlined in Section 1.2, a generalized model that achieves high performance across many different datasets, particularly unseen data, is crucial in biometric verification systems. There are many methods to augment existing deepfake detectors. One model aims to reduce the influence of learned identities of the people in its training dataset, which does not translate to unseen testing data [20]. Other approaches include running adversarial attacks to challenge the detector [21] and using few-shot tuning on previously unseen test datasets [22]. However, many of these methods involve heavy implementation and augmentation to an existing model. It is unclear which methods will be most effective for a specific detector, and how these methods will be able to adapt as more deepfake generation tools become available to the public.

Ensemble models are one way to improve adaptability to novel deepfake generation methods [23] without significant implementation or retraining effort. Ensembles have been used in many image classification tasks, including generating uncertainties for active learning on unlabeled data [24], medical image classification [25], and object detection [26]. Rokach [27] reviews many different methods to combine multiple classifiers, but for this work we focus on random forest classifiers. These models use a large number of decision trees to generate many classification predictions and settle on a final classification through majority vote [27].

Chapter 3

Methods

3.1 Datasets

To evaluate our ensemble model on deepfakes generated on a wide variety of methods, we collect six datasets that collectively cover the most common methods of deepfake generation: latent diffusion, variational autoencoders (VAEs), and GANs. Thank you to Rohan Wagh for collecting the FaceForensics++, ProGAN, StarGAN, and WhichFace datasets described below.

The DeepFakeFace dataset [28] focuses on deepfake images produced by latent diffusion. It includes 30,000 real face images from the IMDB-WIKI dataset. We randomly select 500 deepfake faces generated through Stable Diffusion v1.5 and 500 deepfake faces generated through Stable Diffusion Inpainting to include in our overall dataset. DeepFakeFace [28] includes thousands of images created through InsightFace, an open-source library and commercial product for face detection, alignment, and swapping. We randomly sample 500 InsightFace deepfakes for our overall dataset.

Our second dataset, FaceForensics++ [29], contains deepfakes generated from four methods: Face2Face, FaceSwap, DeepFakes, and NeuralTextures. FaceSwap and DeepFakes are two models that extract a face from a source video and transfers it to a target video, while

Table 3.1: Deepfake face image sources in overall dataset

Family	**Dataset**	**Year**	**Method**	**Number**
Diffusion	DeepFakeFace	2023	Stable Diffusion v1.5 Inpainting InsightFace	500 500 500
GAN	Individual ProGAN StarGAN WhichFace	2024 2017 2018 2019	FaceswapGAN ProgressiveGAN StarGAN StyleGAN	1500 200 2000 1000
VAE	FaceForensics++	2019	Deepfakes	2698

Face2Face and NeuralTextures specialize in facial reenactment. In our dataset, we only include part of the DeepFakes images, which are generated through VAEs.

Due to the rapid advances and popularity of GANs, we include several of the most popular GANs in our dataset. The Individualized Deepfake Detection Dataset (Individual) [30] uses FaceswapGAN [31] to reconstruct real images from the CelebDFv2 dataset. We partition a set of 1500 real images and 1500 deepfake images to include in this dissertation's overall dataset. Progressive GAN (ProGAN) [32] was developed in 2017 and adds additional layers that increase the detail of the generator and discriminator as they get further into training. StarGAN [33] was developed in 2018 and built for facial attribute and expression transfer, making it likely to be used in tools for generating deepfakes. StyleGAN [5] uses an intermediate latent space, allowing the GAN to learn general attributes and perform better at transferring features from source to target images. This GAN is widely used in popular deepfake generation technologies like whichfaceisreal.com [34], from which we have taken 1000 deepfakes and included in our overall dataset as the WhichFace portion.

3.2 Base Detector Model Selection

We begin our ensemble model by selecting the base classifiers from existing deepfake detection models. We identify three models that cover all the main deepfake generation methods and

Table 3.2: Real face image sources in overall dataset

Family	**Dataset**	**Year**	**Method**	**Number**
Diffusion	DeepFakeFace	2023	IMDB-WIKI	1500
GAN	Individual	2024	CelebDFv2	1510
	ProGAN	2017	CelebA	200
	StarGAN	2018	CelebA	2000
	WhichFace	2019	FFHQ	1000
VAE	FaceForensics++	2019	Deepfakes	2698

Table 3.3: Overview of the models used in this study and their training data

Model Name	**Year**	**Training Data Family**	**Methods**
MesoNet [35]	2018	VAE	Deepfake Face2Face
DCT [36]	2022	GAN	ProGAN, StyleGAN, ProjectedGAN Diff-StyleGAN2, Diff-ProjectedGAN
		Diffusion	DDPM, IDDPM, ADM, PNDM, LDM
Dolos [37]	2024	Diffusion	Repaint–P2, Repaint–LDM LaMa, Pluralistic GAN

bring a unique approach to deepfake detection. Table 3.3 lists all the methods that were used to create training deepfake images for each model. The following sections provide an overview of each model's methods and architecture.

3.2.1 Discrete Cosine Transform (DCT)

Many GANs produce frequency artifacts that can be used to differentiate the model's generated images from real images. Applying frequency transforms like discrete Fourier transforms (DFT) and discrete cosine transforms (DCT) before using logistic regression to classify shows significant improvements for diffusion-generated images, although performance still lags behind that for GAN-generated images [36]. In this dissertation, we choose to use the DCT, whose formula given a grayscale image I with height H and width W is displayed in Eq 3.1, with logistic regression from Ricker et al. [36] as one of our base classifiers.

$$I'[k,l] = p_k p_l \sum_{x=0}^{H-1} \sum_{y=0}^{W-1} I[x,y] \cos\left[\frac{\pi}{H}\left(x+\frac{1}{2}\right)k\right] \cos\left[\frac{\pi}{W}\left(y+\frac{1}{2}\right)l\right] \tag{3.1}$$

$$p_k = \begin{cases} \sqrt{\frac{1}{H}} & k = 0 \\ \sqrt{\frac{2}{H}} & 1 \leq k \leq H-1 \end{cases} \tag{3.2}$$

$$p_l = \begin{cases} \sqrt{\frac{1}{W}} & l = 0 \\ \sqrt{\frac{2}{W}} & 1 \leq l \leq W-1 \end{cases} \tag{3.3}$$

We also create two versions of the model with different compression factors, DCT(0.1) and DCT(0.5). This allows us to explore how an ensemble is improved or hurt when adding two very similar models, with the same architecture but different parameters.

With this model we add frequency-based deepfake detection to our evaluation and ensemble model, which our other two models do not cover. One main question is whether DCT can maintain its strong performance on GAN-generated images while also being generalized to perform well on deepfake images generated through diffusion or VAEs, which may not have similar frequency artifacts that GAN-generated images have. Our proposed ensemble model aims to answer this generalization problem by relying on additional models with different training data while heavily weighing DCT's predictions for deepfakes generated through a GAN.

3.2.2 Dolos

We evaluate the "Patches" architecture category described by the authors of the Dolos model [37], which is based on the Patch-Forensics model [38]. This localization model truncates the Resnet and Xception models to obtain binary deepfake predictions on small regions of an image. All the binary patch classifications are averaged to produce a final classification for the image as a whole. The authors of Dolos retrain a version of the Patch-Forensics model that uses two layers of Xception [39] model blocks with diffusion-generated deepfake images. They experiment with three setups that provide the model with varying amounts of information. For this dissertation, we choose the model weights from Setup B, where images are partially manipulated but only classified with real or fake, with no further localization details. This setup is closest to our context of biometric verification systems: most deepfake attacks manipulate the face without changing the image background, and any detection system should classify images as fake if any part has been manipulated.

We select Dolos as it is trained on diffusion-generated deepfake images, providing insight on how training data affects model performance across a broad dataset. Including Dolos in our ensemble model introduces a new deepfake detection strategy that takes advantage of differences between deepfaked image portions and the background, which is often not manipulated.

3.2.3 MesoNet

MesoNet focuses on detecting deepfakes that have been highly compressed or have low quality [35]. This model uses a small number of layers to learn the intermediate properties of deepfake images, rather than learn on the pixel or patch level. We evaluate the Meso-4 network, which starts with four convolutional and pooling layers before using two fully-connected layers to produce a classification. Further analysis of the neuron activation weights reveals that highly-detailed facial features like eyes pushed the classification towards real, while a detailed background and comparatively blurry face pushed the classification towards deepfake [35].

MesoNet brings a focus on VAE-generated deepfakes and low-quality images, which the two previous models lack. In addition, MesoNet is a larger neural network with more layers and parameters compared to DCT and Dolos. Adding this as a base model diversifies the model architectures that we explore in the ensemble combinations in Section 3.4. These aspects allow us to further explore ensemble generalization across the three main architectures for deepfake generation.

3.3 Base Model Evaluation

Before evaluating ensemble model architectures, we individually test each base model outlined in Section 3.2 on our overall dataset, which is described in Section 3.1. For all four models DCT(0.1), DCT(0.5), Dolos, and MesoNet, we use pretrained weights provided by the models' respective authors rather than retraining on our dataset, to evaluate how well these models generalize to unseen data. All models are modified to return positive (1) for deepfake predictions, and negative (0) for real predictions. Since all four models return a probability between 0 and 1, we use each model's ROC curve to find the best threshold for the overall dataset. Accuracy, precision, and recall are all calculated using each model's best threshold. Any probability less than the threshold was classified as a real prediction, and any probability greater than the threshold was classified as a deepfake prediction.

Table 3.4 contains each model's accuracy, precision, recall, and AUC on our overall dataset, as well as the same metrics for each of our six data sources. Thank you to Rohan Wagh for evaluating the DCT models, calculating all thresholds for the four base models, and generating the metrics presented in this table.

Both versions of DCT, DCT(0.1) and DCT(0.5), significantly outperform Dolos and MesoNet on all overall metrics, particularly in total accuracy and precision. DCT is very effective for GAN-generated deepfakes, with both version achieving at least 90% accuracy on ProGAN, StarGAN, and WhichFace. Comparatively, DCT does not perform well to deepfakes generated through diffusion, represented through the DeepFakeFace data. Both DCT(0.1) and DCT(0.5) perform slightly above chance in accuracy and have very low recall.

Although Dolos performs at chance for five out of our six source datasets, its performance on DeepFakeFace stands out. Its total accuracy and recall are significantly higher than DCT(0.1), DCT(0.5), and Mesonet. In the context of biometric verification systems, a false positive is much less harmful than a false negative, as the latter may allow a malicious actor past the identity verification. Thus, even though Dolos does not achieve the highest precision on DeepFakeFace, we prioritize recall over precision. Dolos' performance correlates with how the model was trained mainly on deepfakes generated through latent diffusion. It performs well on our only diffusion deepfake dataset, and performs poorly on all our VAE- and GAN-generated datasets.

Table 3.4: Base model results for overall and individual datasets. Metrics include total accuracy (Acc), deepfake accuracy (Fake), real accuracy (Real), precision (Prec), recall (Rec), and area under the ROC curve (AUC). *indicates high performance as benchmark dataset was in the models training data

Model	**Dataset**	**Metrics**					
		Acc	**Fake**	**Real**	**Prec**	**Rec**	**AUC**
DCT(0.1) threshold = 0.001	DeepFakeFace	54.3	10.9	97.7	82.8	10.9	77.8
	FaceForensics++	67.5	42.4	92.5	**88.9**	44.4	85.2
	Individual	**65.6**	43.4	81.7	70.2	43.4	69.3
	ProGAN*	**100.0**	100.0	100.0	**100.0**	**100.0**	100.0
	StarGAN	**95.2**	93.2	97.2	**97.1**	93.2	98.8
	WhichFace	98.8	98.5	99.0	99.0	98.5	99.9
	Overall	**72.2**	58.3	86.1	80.7	**58.3**	77.8
DCT(0.5) threshold = 0.001	DeepFakeFace	51.0	3.1	99.5	**83.0**	2.6	77.1
	FaceForensics++	54.6	14.8	94.4	72.3	14.8	64.4
	Individual	56.9	23.3	90.3	**70.6**	23.3	65.8
	ProGAN*	**100.0**	100.0	100.0	**100.0**	**100.0**	100.0
	StarGAN	91.7	89.4	94.0	93.7	89.4	97.1
	WhichFace	**99.3**	99.2	99.4	**99.4**	**99.2**	100.0
	Overall	68.6	42.8	94.3	**88.2**	42.8	76.9
Dolos threshold = 0.508	DeepFakeFace	**68.1**	86.7	49.4	63.2	**86.7**	71.3
	FaceForensics++	50.1	0.0	99.9	0.5	0.0	19.8
	Individual	50.8	35.1	66.4	50.9	35.1	50.5
	ProGAN	50.0	0.0	100.0	50.0	**100.0**	27.4
	StarGAN	50.0	0.0	100.0	50.0	**100.0**	0.9
	WhichFace	50.0	100.0	0.0	0.0	0.0	34.1
	Overall	52.9	14.9	90.7	61.6	14.9	23.9
MesoNet threshold = 0.420	DeepFakeFace	53.2	27.8	78.5	56.4	27.8	54.8
	FaceForensics++	**68.7**	74.8	62.5	66.6	**74.8**	74.6
	Individual	61.1	64.7	57.4	60.2	**64.7**	64.7
	ProGAN	53.5	21.5	85.5	59.7	21.5	49.7
	StarGAN	51.8	67.0	36.7	51.4	67.0	50.6
	WhichFace	51.4	68.3	34.5	51.0	68.3	50.7
	Overall	57.3	39.8	74.7	61.1	39.8	57.3

MesoNet achieves the highest accuracy and recall on the FaceForensics++ portion of our dataset, which consists of VAE-generated deepfake images. This follows from MesoNet's training data also being generated through VAEs. The model also performs well on the Individual dataset; DCT(0.1) achieves the best total accuracy but MesoNet has significantly higher recall. On the other GAN- and diffusion-generated datasets, MesoNet's accuracy is close to 50%.

These results indicate concerns about generalization in individual detector models. All four of our base models have strong performance on test data that comes from the same generation family, likely due to these images sharing common distributions, attributes, or artifacts. However, when presented with test data outside their training data distribution, we saw large drops in performance, particularly in total accuracy and AUC. In deepfake detection applications like biometric verification systems, generalization is critical since a system must defend against all the deepfake generation tools that malicious actors have access to. In Section 3.4, we improve generalization by combining the various strengths of multiple detection models into an ensemble detector.

3.4 Ensemble Model Design and Implementation

Although there are many ensemble methods, we decided to use the random forest method because this model performed the best in initial exploratory experiments. We combine predictions from multiple base models into one ensemble model using a random forest classifier made up of several decision trees. Each tree is created to classify a random sample of the training data. Our random forest classifier's trees decide the best split at each tree node by minimizing the Gini impurities of the resulting child nodes, weighed by the number of images in each child. The Gini impurity formula is defined as

$$I_G = 1 - (p_d^2 + p_r^2) \tag{3.4}$$

with p_d representing the proportion of deepfake images at a given node and p_r representing the proportion of real images. The best split for a decision tree node occurs when one child consists entirely of deepfake images and the other child consists entirely of real images, thus achieving a Gini impurity of 0 for both child nodes.

After hyperparameter tuning to achieve the highest overall accuracy, we settled on a random forest classifier with nine decision trees, where each tree has a maximum depth of three and a node requires at least five samples to split. The random forest classifier receives class probabilities (probability of an image being a deepfake, and probability of an image being real) from individual decision trees and outputs a real/deepfake prediction based on which class has the higher average probability. We define this function as

$$\text{Ensemble Prediction} = \operatorname{argmax}\left(\frac{1}{n}\sum_{n=1}^{9} p_c(t_n)\ \forall\ c \in \{0,1\}\right)$$

where 0 represents a deepfake prediction, 1 represents a real prediction, $p_0(t_n)$ represents the deepfake probability output by decision tree n, and $p_1(t_n)$ represents the real probability output by decision tree n.

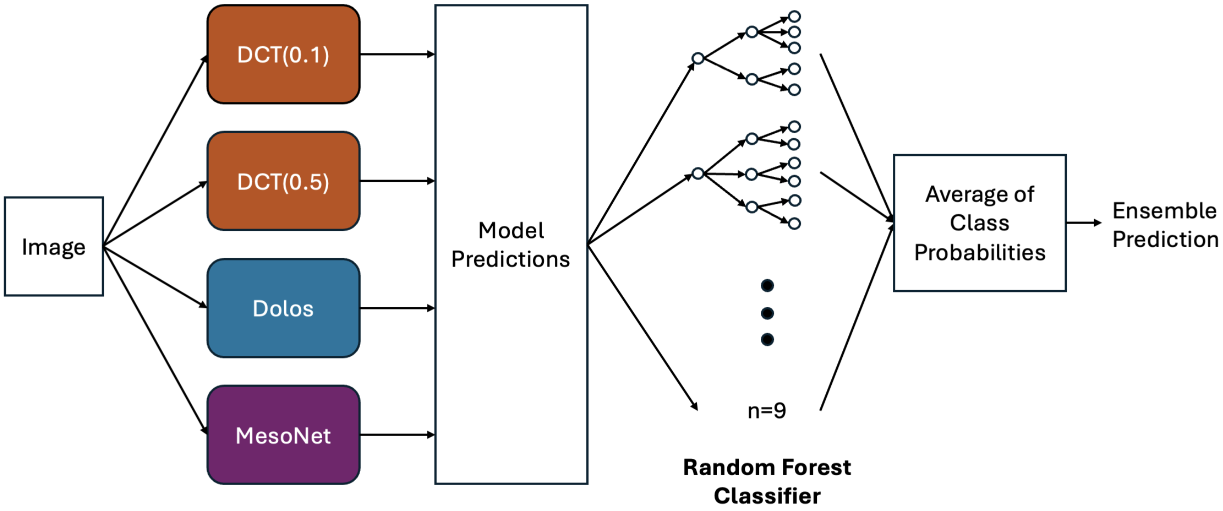


Figure 3.1: Architecture of the four-model ensemble

Figure 3.1 shows the architecture of the ensemble composed of all four base models. We used the same classifier architecture to experiment with all possible combinations of base models in pairs, groups of three, and all four models together. Running an exhaustive list of ensembles allowed us to investigate whether adding more models can negatively impact performance as poor predictions gain more weight.

To train the random forest classifier, we split our overall dataset into 80% training data and 20% ensemble testing data. We use five-fold cross-validation to measure how stable ensemble models are across multiple runs. Results are presented in Section 4.1.

Chapter 4

Results

4.1 Evaluation of Exhaustive Ensemble Combinations

In total, we evaluate six ensembles of two base models, four ensembles of three base models, and one ensemble of all four base models that we investigated in Section 3.2. Total accuracy, precision, and recall on our overall dataset are presented in Table 4.1 for each ensemble model version. For each metric, we display the mean and standard deviation across five training runs, each using a different fold of our overall dataset.

All ensemble versions perform at least as well as the top individual base model DCT(0.1), and improve on DCT(0.1)'s overall accuracy of 72.2% by up to 7%. The ensemble with all four base models does perform the best out of all ensemble versions, as it has the highest accuracy and recall as well as one of the highest precision percentages. This may suggest that adding more base models is the best way to improve ensemble performance. However, comparing the DCT(0.1) & MesoNet combination with the DCT(0.1) & and DCT(0.5) & MesoNet combination shows that this is not always the case. Adding DCT(0.5), which is very similar to the DCT(0.1) model already included in the initial combination, does not improve accuracy and slightly decreases recall.

Analyzing the ensemble combinations that produce the second and third highest accuracies,

Table 4.1: Performance comparison of different ensemble models. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Ensemble Model Combination	**Acc**	**Prec**	**Rec**
DCT(0.1)	72.2	80.7	58.3
DCT(0.5)	68.6	**88.2**	42.8
Dolos	52.9	61.6	14.9
MesoNet	57.3	61.1	39.8
DCT(0.1) & DCT(0.5)	72.1 ± 0.8	78.4 ± 1.6	61.2 ± 3.5
DCT(0.1) & Dolos	75.5 ± 0.8	77.9 ± 3.0	71.5 ± 3.5
DCT(0.1) & MesoNet	76.3 ± 0.2	81.6 ± 0.6	68.0 ± 1.0
DCT(0.5) & Dolos	73.9 ± 0.4	83.1 ± 2.2	60.1 ± 2.4
DCT(0.5) & MesoNet	72.7 ± 0.7	75.5 ± 2.3	67.2 ± 2.6
Dolos & MesoNet	72.6 ± 0.8	74.8 ± 1.5	68.2 ± 4.7
DCT(0.1) & DCT(0.5) & Dolos	74.4 ± 1.8	80.4 ± 1.4	64.6 ± 4.4
DCT(0.1) & DCT(0.5) & MesoNet	75.4 ± 0.8	81.8 ± 0.2	65.3 ± 2.2
DCT(0.1) & Dolos & MesoNet	78.0 ± 1.9	84.1 ± 3.0	69.1 ± 3.2
DCT(0.5) & Dolos & MesoNet	77.0 ± 1.7	81.6 ± 2.9	70.1 ± 5.6
DCT(0.1) & DCT(0.5) & Dolos & MesoNet	**79.2 ± 1.5**	83.4 ± 2.0	**72.9 ± 2.0**

DCT(0.1) & Dolos & MesoNet and DCT(0.5) & Dolos & MesoNet respectively, reveal how important a diverse set of base models can be. Both groups contain only one version of DCT along with Dolos and MesoNet, ensuring that the ensemble has a base model trained on all the three main deepfake generation methods.

Combining our base models into an ensemble also significantly improves our recall. The highest recall achieved by an individual base model was 58.3% by DCT(0.1). The four-model ensemble increases recall by more than 14%, reducing the number of deepfake images that are mistakenly classfied as real and preventing malicious actors from gaining access to sensitive systems. This is another positive benefit that comes from using multiple base models trained to detect different types of deepfakes. A model like DCT, which is strongest at detecting artifacts in GAN-generated images, would miss many deepfakes generated through diffusion or VAEs. However, an ensemble model can weigh positive predictions from other base models more heavily than a negative prediction from DCT, thus catching novel deepfake generation methods. Interestingly, the DCT(0.1) & Dolos ensemble achieves a slightly higher recall than DCT(0.1) & Dolos & MesoNet or DCT(0.5) & Dolos & MesoNet.

4.2 Building Ensemble Models Based on Accuracy

We now analyze different ensemble combinations to come up with a framework for selecting base models. Especially when working with a large selection of base models all with different strengths and training data, it is important to select high-performing ensembles without an exhaustive search. We begin by selecting a pair of base models where, for as many of our six data sources as possible, one of the models has the highest accuracy or close to the highest accuracy. From Table 3.4 we select DCT(0.1) and Dolos. Even though neither model has the highest accuracy for the FaceForensics++ and WhichFace datasets, DCT(0.1) is within 1-2% of the top accuracy for both of these datasets. In Figure 4.1, we compare the overall rates of correct predictions, false positives, and false negatives for the DCT(0.1) & Dolos ensemble as well as the individual base models. While the ensemble has a lower true negative rate and is less accurate at classifying real images, its ability to detect true positive deepfake images has significantly improved. The accuracy, precision, and recall for DCT(0.1) & Dolos broken down across datasets is presented in Table 4.2. Comparing these results against DCT(0.1) and Dolos individually in Table 3.4, we see that the ensemble successfully incorporates Dolos' strong performance in diffusion-generated images (DeepFakeFace) without a large degradation in DCT(0.1)'s strong performance on GAN-generated images.

In Table 4.2, we also compare the DCT(0.1) & Dolos ensemble against the ensemble with the highest accuracy, DCT(0.1) & DCT(0.5) & Dolos & MesoNet. The smaller ensemble DCT(0.1) & Dolos falls slightly behind the four-model ensemble in all metrics. Both ensembles have similar accuracy across four datasets, but DCT(0.1) & DCT(0.5) & Dolos & MesoNet has significantly better accuracy for StarGAN and WhichFace. Even though combining Dolos with DCT(0.1) improved performance on diffusion-generated deepfakes, it weakened

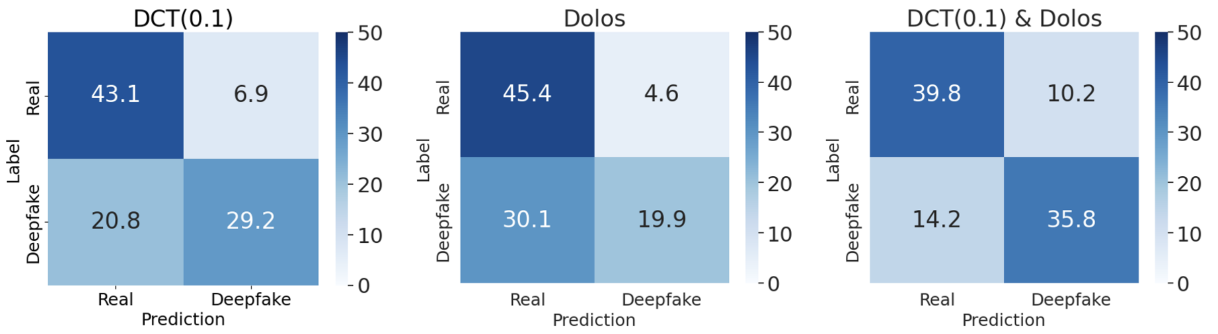


Figure 4.1: Confusion matrices for the individual DCT(0.1) and Dolos models and their ensemble

DCT(0.1)'s strong performance on GAN-generated deepfakes.

However, amongst the two-model ensembles, DCT(0.1) & Dolos achieves the highest recall and is within 1% of the best accuracy from DCT(0.1) & MesoNet. We include the metrics for DCT(0.1) & MesoNet to Table 4.2 in order to compare these two-model ensembles. Without Dolos, accuracy and recall on diffusion-generated images are extremely low, but accuracy on both StarGAN and WhichFace are significantly higher. This leads to the overall accuracy of DCT(0.1) & MesoNet being slightly better than DCT(0.1) & Dolos, although the latter may be considered more well-rounded and suitable for a verification system. Despite the non-optimal performance, selecting base models using their individual accuracies and aiming for coverage across many deepfake generation types is a promising strategy to build an ensemble.

4.3 Building Ensemble Models Based on Precision

We next look at selecting base models based on their individual precision, again aiming for high precision rates across all data sources. Collectively, DCT(0.1) and DCT(0.5) give us the highest precision scores in the six source datasets in our study. The DCT(0.1) & DCT(0.5) ensemble achieves an overal accuracy of 72.1% on average, which is low compared to the other two-model ensembles in Table 4.1. In fact, its overall accuracy is about equal to the overall accuracy of DCT(0.1) in Table 3.4, although overall precision and recall are slightly

Table 4.2: Performance of an accuracy-based ensemble model across all datasets. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Model	**Dataset**	**Metrics** **Acc**	 **Prec**	 **Rec**
DCT(0.1) & Dolos	DeepFakeFace	71.0 ± 4.3	78.2 ± 9.0	62.3 ± 17.4
	FaceForensics++	74.2 ± 2.2	90.7 ± 3.8	54.0 ± 5.4
	Individual	64.2 ± 3.0	68.4 ± 4.8	52.9 ± 3.1
	ProGAN	98.2 ± 1.9	96.7 ± 3.4	100.0 ± 0.0
	StarGAN	81.2 ± 1.9	72.9 ± 2.0	99.3 ± 0.6
	WhichFace	87.0 ± 3.8	79.8 ± 4.5	99.5 ± 1.1
	Overall	**75.5 ± 0.8**	**77.9 ± 3.0**	**71.5 ± 3.5**
DCT(0.1) & MesoNet	DeepFakeFace	52.3 ± 1.2	84.1 ± 12.6	5.5 ± 2.1
	FaceForensics++	76.0 ± 1.3	81.0 ± 2.3	68.0 ± 2.8
	Individual	67.9 ± 1.5	68.2 ± 2.7	66.9 ± 2.9
	ProGAN	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	StarGAN	89.4 ± 1.6	84.3 ± 2.9	97.0 ± 1.2
	WhichFace	94.9 ± 1.1	92.0 ± 1.9	98.4 ± 1.3
	Overall	**76.3 ± 0.2**	**81.6 ± 0.6**	**68.0 ± 1.0**
DCT(0.1) & DCT(0.5) & Dolos & MesoNet	DeepFakeFace	70.6 ± 7.5	81.5 ± 4.3	54.9 ± 23.0
	FaceForensics++	76.8 ± 3.3	90.4 ± 2.5	60.0 ± 7.8
	Individual	66.9 ± 1.4	70.1 ± 1.6	58.7 ± 5.6
	ProGAN	99.0 ± 0.6	98.0 ± 1.1	100.0 ± 0.0
	StarGAN	88.8 ± 4.2	82.9 ± 5.8	98.5 ± 0.5
	WhichFace	94.1 ± 1.0	89.6 ± 1.7	99.7 ± 0.4
	Overall	**79.2 ± 1.5**	**83.4 ± 2.0**	**72.9 ± 2.0**

higher. Figure 4.2 compares our ensemble against the existing DCT(0.1) and DCT(0.5) models individually, revealing that although the ensemble has the highest true positive rate, this increase is not enough to offset the decrease in true negatives.

We can continue to analyze the ensemble's performance on specific deepfake generation methods through Table 4.3, which displays the DCT(0.1) & DCT(0.5) ensemble model's metrics for the overall dataset as well as the six portions. Comparing against DCT(0.1)'s individual performance in Table 3.4, the two models perform very similarly on most of the datasets. However, the DCT(0.1) & DCT(0.5) ensemble does noticably worse on the StarGAN and WhichFace datasets, particularly in precision. The increase in false positives suggests

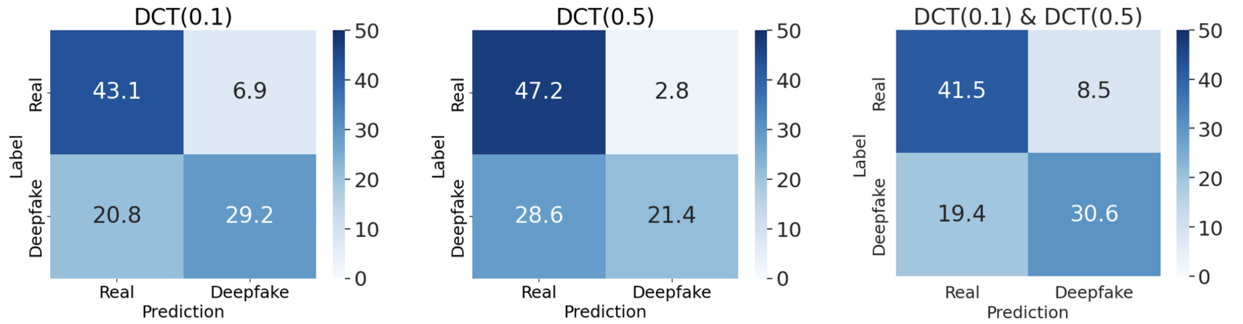


Figure 4.2: Confusion matrices for the individual DCT(0.1) and DCT(0.5) models and their ensemble

that combining two very similar models causes the ensemble to place too much weight on the positive predictions coming from either model.

Table 4.3: Performance of a precision-based ensemble model across all datasets. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Model	**Dataset**	**Metrics** **Acc**	 **Prec**	 **Rec**
DCT(0.1) & DCT(0.5)	DeepFakeFace	54.9 ± 0.2	81.7 ± 4.6	12.7 ± 1.1
	FaceForensics++	69.3 ± 3.6	81.6 ± 2.9	49.7 ± 8.5
	Individual	63.2 ± 2.1	68.3 ± 2.0	48.8 ± 5.3
	ProGAN	99.3 ± 1.1	98.6 ± 2.1	100.0 ± 0.0
	StarGAN	83.4 ± 3.0	75.6 ± 3.2	98.9 ± 0.4
	WhichFace	91.1 ± 1.6	84.9 ± 2.2	100.0 ± 0.0
	Overall	**72.1 ± 0.8**	**83.1 ± 2.5**	**61.2 ± 3.5**

While more work needs to be done to analyze why combining two similar base detectors leads to overprediction of deepfakes and whether this pattern generalizes, our experiments highlight the dangers of combining too many models in one ensemble.

4.4 Building Ensemble Models Based on Recall

Lastly, we build an ensemble based on the recall of the individual base models, again aiming to have for each data source at least one base model that achieves high recall on it. We select DCT(0.1) for its high recall on GAN-generated datasets and comparatively high overall recall, Dolos for its high recall on the diffusion-generated dataset, and MesoNet for having the best recall on the FaceForensics++ and Individual datasets. Although DCT(0.5) has the highest recall on the WhichFace dataset, we exclude it since DCT(0.1) already has a recall of 98.5%. Table 4.4 shows metrics for the DCT(0.1) & Dolos & MesoNet on the overall dataset, as well as each of the six source datasets.

Table 4.4: Performance of a recall-based ensemble model across all datasets. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Model	**Dataset**		**Metrics**	
		Acc	**Prec**	**Rec**
DCT(0.1) & Dolos & MesoNet	DeepFakeFace	60.9 ± 5.6	82.6 ± 7.6	26.7 ± 13.5
	FaceForensics++	78.6 ± 2.4	91.7 ± 4.3	63.0 ± 3.7
	Individual	67.0 ± 1.7	70.3 ± 1.1	58.4 ± 4.1
	ProGAN	99.7 ± 0.6	99.5 ± 1.1	100.0 ± 0.0
	StarGAN	90.4 ± 3.7	84.6 ± 5.4	99.3 ± 0.3
	WhichFace	89.6 ± 3.6	83.6 ± 5.1	98.9 ± 1.3
	Overall	**78.0 ± 1.9**	**84.1 ± 3.0**	**69.1 ± 3.2**

This three-model ensemble has one of the highest accuracies amongst all the ensemble combinations in Table 4.1. It also performs well on recall, although the four-model ensemble DCT(0.1) & DCT(0.5) & Dolos & MesoNet improves recall by nearly 4%. This experiment shows the promise of building ensemble models through a recall-based process. In choosing models that minimize false negatives across a wide variety of datasets, we naturally select a model specialized towards each main deepfake generation method: DCT(0.1) across all GAN-generated datasets, MesoNet on the VAE-generated deepfakes, and Dolos on our diffusion-generated dataset. These three models were each trained to detect a different type of deepfake generation, enabling them to avoid incorrectly classifying deepfakes of that type as real images. Even though the models do not perform well on deepfakes outside one family of generation methods, combining them allows the ensemble to weigh positive predictions more and reduce false negatives.

4.5 Training Data Ablation

An important secondary question that this dissertation investigates is the importance of diversity in training data. We perform an ablation study with all variants of our ensemble models where we track accuracy as large subsets of training data are removed. According to Table 3.1, our dataset includes three categories based on deepfake generation family: diffusion, GAN, and VAE. The diffusion category includes the real and fake images from the DeepFakeFace dataset, the VAE category includes the real and fake images from the FaceForensics++ dataset, and the GAN category includes all real and fake images from the remaining four datasets (Individual, ProGAN, StarGAN, WhichFace). For each of these categories, we either (1) randomly remove half of the real and deepfake images in the ensemble's training data, or (2) remove all real and deepfake images that fall under the category. We maintain the same ensemble architecture and evaluation process as described in Section 3.4, and do not remove any images from the test dataset. Table 4.5 shows the accuracies across our ablation study for selected ensemble model versions.

The ensemble model versions maintained similar levels of accuracy for the Half ablation tests. For all three categories, accuracy dropped by less than 1%, with the GAN category having a particularly low degradation likely because it had the largest number of training images. However, completely excluding one category of training images leads to significant performance losses. The None ablation tests for GAN and VAE training data show that accuracy drops by 4.4% and 3.6% respectively. In the Diffusion image category, average ensemble accuracy drops by 0.3% as we move from the Half to the None test, a comparatively small drop due to the smaller size of diffusion-generated images in our dataset.

Ensembles can be more vulnerable to certain ablations based on the characteristics of their base models. One example is the Dolos & MesoNet ensemble as this combination does not include a base model that is strong in detecting GAN-generated images. As the GAN-generated training data is entirely removed, accuracy sharply drops by 7.5%, from

Table 4.5: Overall accuracy and standard deviation across five runs of selected ensemble models, with various subsets of training data removed. The Full column lists the ensemble's accuracy given the entire training dataset available; the Half and None column labels refer to how many images from a category were kept in the ensemble's training dataset. The last row lists the change in accuracy compared to the Full test, averaged across all ensemble model versions.

Model	**Full**	**Diffusion**		**GAN**		**VAE**	
		Half	**None**	**Half**	**None**	**Half**	**None**
DCT(0.1) & DCT(0.5)	72.1 ± 0.8	72.3 ± 0.7	72.1 ± 1.0	72.0 ± 0.8	70.1 ± 0.6	71.2 ± 0.8	69.3 ± 0.5
DCT(0.1) & Dolos	75.5 ± 0.8	73.9 ± 0.6	72.8 ± 1.0	75.7 ± 1.2	73.9 ± 1.6	73.3 ± 0.8	72.3 ± 0.3
DCT(0.1) & MesoNet	76.3 ± 0.2	76.1 ± 0.3	75.9 ± 1.2	74.9 ± 1.9	70.2 ± 0.9	75.8 ± 0.4	73.8 ± 1.2
DCT(0.5) & Dolos	73.9 ± 0.4	70.6 ± 0.7	71.0 ± 0.7	74.2 ± 0.9	69.7 ± 1.8	72.9 ± 0.2	72.6 ± 0.3
DCT(0.5) & MesoNet	72.7 ± 0.7	72.7 ± 0.7	72.5 ± 0.6	72.7 ± 0.7	65.2 ± 4.8	72.1 ± 0.7	71.0 ± 0.6
Dolos & MesoNet	72.6 ± 0.8	71.7 ± 0.7	71.6 ± 0.7	72.4 ± 1.7	64.9 ± 0.6	71.6 ± 0.7	68.5 ± 1.5
DCT(0.1) & DCT(0.5) & Dolos	74.4 ± 1.8	74.0 ± 0.5	72.3 ± 0.6	75.5 ± 1.0	72.4 ± 1.8	73.2 ± 1.0	71.7 ± 1.5
DCT(0.1) & DCT(0.5) & MesoNet	75.4 ± 0.8	75.6 ± 0.7	75.4 ± 0.8	75.0 ± 0.9	70.7 ± 1.2	75.1 ± 1.1	72.4 ± 1.5
DCT(0.1) & Dolos & MesoNet	78.0 ± 1.9	77.1 ± 0.9	77.4 ± 0.6	76.8 ± 1.5	75.1 ± 1.6	77.8 ± 1.4	73.1 ± 1.5
DCT(0.5) & Dolos & MesoNet	77.0 ± 1.7	75.9 ± 1.6	76.2 ± 0.6	78.3 ± 3.1	70.8 ± 4.9	75.2 ± 1.8	71.1 ± 1.3
DCT(0.1) & DCT(0.5) & Dolos & MesoNet	79.2 ± 1.5	77.7 ± 1.2	77.5 ± 0.8	77.2 ± 2.0	76.2 ± 0.6	79.4 ± 1.5	72.2 ± 1.6
Difference from Full		-0.8	-1.1	-0.2	-4.4	-0.9	-3.6

72.4% to 64.9%. In contrast, when the diffusion-generated or VAE-generated training images are removed, the Dolos & MesoNet ensemble sees a moderate loss in accuracy that is similar to other ensemble versions. This example underscores the importance of having base models trained on a diverse set of generation methods. However, the decreases in accuracy for other ensemble combinations that include a GAN-specialized base model highlight that building a diverse training dataset for the ensemble is necessary to achieve the best performance.

Chapter 5

Discussion

5.1 Proposed Framework for Ensemble Development

The approach we have defined throughout this thesis is a practical solution for the generalization challenges that current deepfake detectors face in the short term. Biometric verification systems must constantly respond to improvements in existing deepfake generation tools, as well as novel generation algorithms that are emerging into the public domain. It is critical for these systems to respond to these threats quickly and ensure that there are no false negatives. Ensemble models can easily be adapted for new deepfake generation types by adding an additional base model that has been trained on the new type and achieves high performance on it. Even if the additional model does not perform well on existing generation methods, the ensemble will still achieve high performance through its other base models. Figure 5.1 illustrates this phenomenom with Dolos, our base model that performs significantly better than DCT and MesoNet on diffusion-generated deepfakes, but performs at chance on all other deepfake types. Adding Dolos into two ensembles increased accuracy on diffusion deepfakes by 12% and 8% without any decreases in accuracy for GAN and VAE deepfakes. Incorporating new models can solve major weaknesses in a biometric verification system with comparatively small negative effects on the other base models.

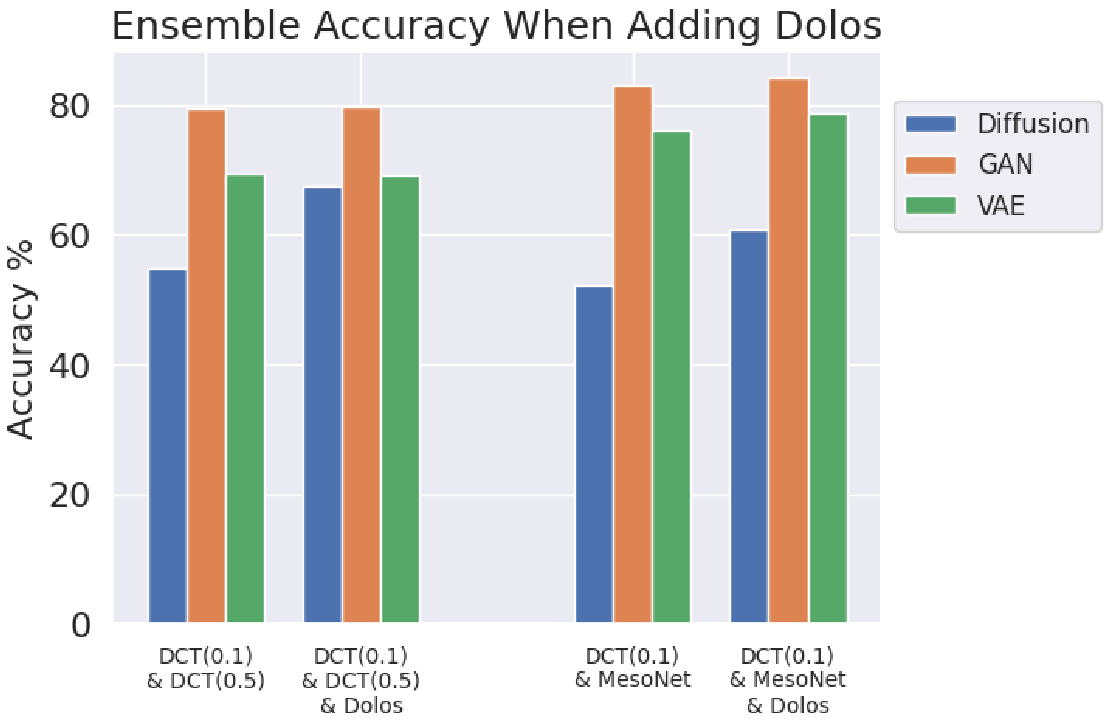


Figure 5.1: Accuracy across deepfake generation families for ensembles with and without the Dolos base model

A large variety of base models is crucial. In this thesis, we separate deepfake generation into three main families: GAN, VAE, and diffusion. We evaluated one model for each deepfake generation family, with two versions of the same model for GAN-generated deepfakes. Our best ensembles include at least one model trained on each of these families. As shown in Figure 5.2, no base model performs well across all three deepfake families, but the DCT(0.1) & DCT(0.5) & Dolos & MesoNet ensemble matches the accuracy of the best model for each generation method. Before beginning to design an ensemble model for a biometric verification system, an evaluation should cover the types of deepfake attacks received, the base models that are available, and how the models perform on a representative sample of attacks.

Based on the approaches discussed in Section 4, we recommend building an ensemble based on the recall of individual base models. This limits the number of false negatives across all deepfake generation types, which is especially relevant for biometric verification systems. A false negative is a much larger security risk in this context as a malicious actor

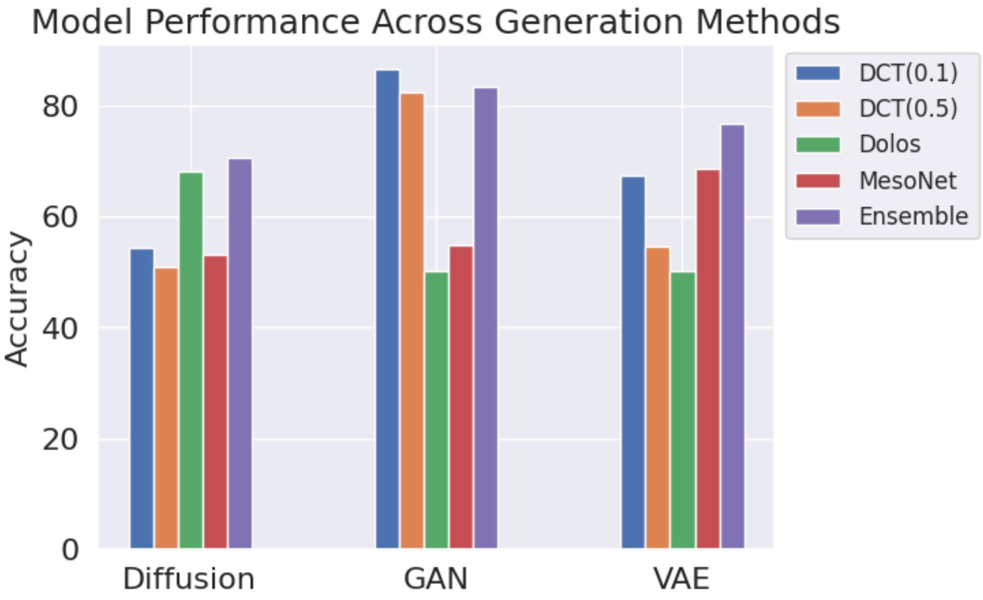


Figure 5.2: Accuracy for all four individual base models and the full four-model ensemble DCT(0.1) & DCT(0.5) & Dolos & MesoNet

may be able to bypass part of a security system, whereas for a false positive, the actual user can reverify their identity and their systems remain protected. Although recall is closely tied to accuracy, accuracy alone does not show whether a model produces mostly false negatives or limits the number of false negatives. As seen in Table 3.4, DCT(0.1)'s accuracy on the FaceForensics++ dataset is very close to MesoNet's accuracy, but DCT(0.1)'s recall is much lower than MesoNet's because the former correctly predicts less than half of the VAE-generated deepfakes.

Once a strong group of base models has been collected, an ensemble can be built by combining the base classifications via a random forest classifier, like in this thesis, or another ensemble method. Training the ensemble requires less time compared to standard deepfake detectors, as these ensembles are only taking several base predictions as input data rather than entire images. The base models themselves do not require any retraining or finetuning, as we no longer need one model to adapt to all the unique and complex deepfakes types in a specific system's dataset. Instead, the ensemble prioritizes having several base models that are strong in a specific deepfake generation type. Once a new generation model gains attention or a weakness is discovered in the existing ensemble, another base model can be selected and the ensemble can be retrained. The amount of time, research effort, and computing resources spent are greatly reduced compared to if the verification system had to retrain its detector or modify the model's architecture.

5.2 Limitations

The size of our dataset and number of models we used were both small, making it difficult to generalize results or identify consistent patterns. In addition, the only two similar base models that we investigated had the same underlying algorithm. Before we can conclude that combining models with the same individual strengths does not improve performance, we need to test with different model architectures that are trained on the same data and achieve similar results to each other across multiple datasets.

Our overall dataset is also unbalanced between the three main deepfake generation families we outline. Of all the deepfake images in our dataset, 17% are diffusion-generated, 30% are VAE-generated, and 53% are GAN-generated. While this often reflects how deepfakes for malicious attacks are generated, an ensemble test that uses an equal portion of all deepfake generation types will be very helpful in understanding how ensembles weigh their base models and how adding additional models can degrade performance.

Looking ahead to real-world implementation, ensemble models take extra computing resources and time when making inferences due to its multiple base models. In this thesis, we do not investigate latency during testing or how much it increases compared to a single detector. However, the tradeoff between increased accuracy and response time will undoubtedly be crucial for individual verification systems to consider.

5.3 Future Work

As a starting point, aspects of the ensemble experiments' implementation can be improved or rerun on a larger scale. Expanding the dataset, adding evaluations for more base models, and including models with different architectures trained on the same deepfake generation families are all next steps that will confirm the results from this thesis. In addition, new ensemble methods beyond random forest classifiers could be explored. Boosting algorithms like AdaBoost begin with weak classifiers but improves on them through weighing misclassifications more heavily. They have been successfully applied in a range of image classification tasks, such as categorizing brain magnetic resonance images for diagnosis [40] and general image classification [41]. Compared to random forest classifiers, AdaBoost may be better suited to learn differences between base detector models, and more able to take advantage of the different strengths that deepfake detectors can offer.

In Section 4.5 we explored reducing the diversity of training data and found that removing a category of deepfake generation entirely led to large performance degradations, but removing half of the data in a category led to minor decreases in accuracy. It is unclear whether a point exists in between these two extremes, where performance drops sharply once we remove more than a specific proportion of training data. The questions of overfitting or not having enough training data are crucial to answer if ensemble deepfake detectors are to be used in a wide variety of applications. This will require an in-depth exploration of different subsets of training data.

As discussed in Section 5.2, more work is needed to analyze how larger ensemble models affect verification system performance. Factors to consider include, but are not limited to, increases in latency and required computing power. Although larger ensemble models can provide higher accuracy and recall, the tradeoffs may not be worthwhile for many systems, which is why we have included Section 5.1 as a framework for constraining the number of base models. Research can also be done on the feasibility of ensembles in applications with a high volume of user authentication requests and what additional restrictions ensembles need to satisfy in these contexts.

Chapter 6

Conclusion

This thesis presents ensemble models as a practical solution for the generalization challenges that currently exist in the field of deepfake detection. We evaluate four models on a varied dataset containing many deepfake generation methods from over the past decade, and find that these models often perform at chance on unseen datasets, particularly ones outside their training data family. We train an exhaustive list of ensemble combinations, and all of our random forest ensembles outperform the leading base model. Comparisons between different ensemble combinations reveal how additional base models can substantially improve performance on a deepfake type, but can degrade the ensemble's strong performance on other types. We propose using the recall and the false negative rates of base models to build the best ensemble model, particularly in biometric verification where false negatives are especially important to avoid. This framework provides a starting point for more work to be done in exploring ensemble deepfake detectors and potential implementations.

Appendix A

Ensemble Performance

Table A.1: Performance of the four-model ensemble across all datasets. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Model	**Dataset**		**Metrics**	
		Acc	**Prec**	**Rec**
DCT(0.1) & DCT(0.5) & Dolos & MesoNet	DeepFakeFace	70.6 ± 7.5	81.5 ± 4.3	54.9 ± 23.0
	FaceForensics++	76.8 ± 3.3	90.4 ± 2.5	60.0 ± 7.8
	Individual	66.9 ± 1.4	70.1 ± 1.6	58.7 ± 5.6
	ProGAN	99.0 ± 0.6	98.0 ± 1.1	100.0 ± 0.0
	StarGAN	88.8 ± 4.2	82.9 ± 5.8	98.5 ± 0.5
	WhichFace	94.1 ± 1.0	89.6 ± 1.7	99.7 ± 0.4
	Overall	**79.2 ± 1.5**	**83.4 ± 2.0**	**72.9 ± 2.0**

Table A.2: Performance of three-model ensembles across all datasets. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Model	**Dataset**	**Metrics**		
		Acc	**Prec**	**Rec**
DCT(0.1) & DCT(0.5) & Dolos	DeepFakeFace	67.4 ± 8.8	84.7 ± 5.5	44.1 ± 25.7
	FaceForensics++	69.1 ± 1.9	86.7 ± 1.9	45.1 ± 3.9
	Individual	62.9 ± 2.2	69.6 ± 4.1	45.6 ± 2.5
	ProGAN	98.8 ± 0.8	97.6 ± 1.7	100.0 ± 0.0
	StarGAN	84.7 ± 1.7	77.0 ± 2.1	99.1 ± 0.6
	WhichFace	91.1 ± 2.2	85.0 ± 3.3	100.0 ± 0.0
	Overall	**74.4 ± 1.8**	**80.4 ± 1.4**	**64.6 ± 4.4**
DCT(0.1) & DCT(0.5) & MesoNet	DeepFakeFace	52.0 ± 1.1	81.4 ± 15.1	5.0 ± 2.3
	FaceForensics++	73.4 ± 2.2	79.8 ± 1.6	62.6 ± 4.9
	Individual	66.2 ± 2.1	67.8 ± 2.5	61.3 ± 3.0
	ProGAN	99.5 ± 1.1	99.0 ± 2.1	100.0 ± 0.0
	StarGAN	90.1 ± 1.7	85.8 ± 2.9	96.4 ± 2.1
	WhichFace	95.6 ± 1.6	92.3 ± 2.9	99.6 ± 0.4
	Overall	**75.4 ± 0.8**	**81.8 ± 0.2**	**65.3 ± 2.2**
DCT(0.1) & Dolos & MesoNet	DeepFakeFace	60.9 ± 5.6	82.6 ± 7.6	26.7 ± 13.5
	FaceForensics++	78.6 ± 2.4	91.7 ± 4.3	63.0 ± 3.7
	Individual	67.0 ± 1.7	70.3 ± 1.1	58.4 ± 4.1
	ProGAN	99.7 ± 0.6	99.5 ± 1.1	100.0 ± 0.0
	StarGAN	90.4 ± 3.7	84.6 ± 5.4	99.3 ± 0.3
	WhichFace	89.6 ± 3.6	83.6 ± 5.1	98.9 ± 1.3
	Overall	**78.0 ± 1.9**	**84.1 ± 3.0**	**69.1 ± 3.2**
DCT(0.5) & Dolos & MesoNet	DeepFakeFace	63.2 ± 11.8	77.3 ± 4.3	37.0 ± 33.0
	FaceForensics++	78.1 ± 4.6	89.6 ± 1.4	63.6 ± 9.7
	Individual	61.7 ± 2.6	63.0 ± 4.9	59.2 ± 10.2
	ProGAN	90.8 ± 11.3	95.0 ± 3.9	85.5 ± 20.8
	StarGAN	93.0 ± 1.8	89.3 ± 1.9	97.7 ± 2.5
	WhichFace	83.4 ± 8.8	77.5 ± 9.3	95.3 ± 5.3
	Overall	**77.0 ± 1.7**	**81.6 ± 2.9**	**70.1 ± 5.6**

Table A.3: Performance of two-model ensembles across all datasets. Metrics include total accuracy (Acc), precision (Prec), and recall (Rec). The mean and standard deviation across 5 runs are reported below.

Model	**Dataset**	**Metrics**		
		Acc	**Prec**	**Rec**
DCT(0.1) & DCT(0.5)	DeepFakeFace	54.9 ± 0.2	81.7 ± 4.6	12.7 ± 1.1
	FaceForensics++	69.3 ± 3.6	81.6 ± 2.9	49.7 ± 8.5
	Individual	63.2 ± 2.1	68.3 ± 2.0	48.8 ± 5.3
	ProGAN	99.3 ± 1.1	98.6 ± 2.1	100.0 ± 0.0
	StarGAN	83.4 ± 3.0	75.6 ± 3.2	98.9 ± 0.4
	WhichFace	91.1 ± 1.6	84.9 ± 2.2	100.0 ± 0.0
	Overall	**72.1 ± 0.8**	**83.1 ± 2.5**	**61.2 ± 3.5**
DCT(0.1) & Dolos	DeepFakeFace	71.0 ± 4.3	78.2 ± 9.0	62.3 ± 17.4
	FaceForensics++	74.2 ± 2.2	90.7 ± 3.8	54.0 ± 5.4
	Individual	64.2 ± 3.0	68.4 ± 4.8	52.9 ± 3.1
	ProGAN	98.2 ± 1.9	96.7 ± 3.4	100.0 ± 0.0
	StarGAN	81.2 ± 1.9	72.9 ± 2.0	99.3 ± 0.6
	WhichFace	87.0 ± 3.8	79.8 ± 4.5	99.5 ± 1.1
	Overall	**75.5 ± 0.8**	**77.9 ± 3.0**	**71.5 ± 3.5**
DCT(0.1) & MesoNet	DeepFakeFace	52.3 ± 1.2	84.1 ± 12.6	5.5 ± 2.1
	FaceForensics++	76.0 ± 1.3	81.0 ± 2.3	68.0 ± 2.8
	Individual	67.9 ± 1.5	68.2 ± 2.7	66.9 ± 2.9
	ProGAN	100.0 ± 0.0	100.0 ± 0.0	100.0 ± 0.0
	StarGAN	89.4 ± 1.6	84.3 ± 2.9	97.0 ± 1.2
	WhichFace	94.9 ± 1.1	92.0 ± 1.9	98.4 ± 1.3
	Overall	**76.3 ± 0.2**	**81.6 ± 0.6**	**68.0 ± 1.0**
DCT(0.5) & Dolos	DeepFakeFace	74.3 ± 1.3	77.4 ± 1.8	68.9 ± 4.8
	FaceForensics++	60.4 ± 2.4	82.5 ± 4.4	26.5 ± 6.7
	Individual	57.7 ± 0.6	67.6 ± 2.5	29.3 ± 3.5
	ProGAN	98.3 ± 1.1	96.7 ± 2.1	100.0 ± 0.0
	StarGAN	93.6 ± 0.9	89.6 ± 1.3	98.6 ± 0.6
	WhichFace	90.2 ± 4.9	84.5 ± 6.7	99.3 ± 0.8
	Overall	**73.9 ± 0.4**	**83.1 ± 2.2**	**60.1 ± 2.4**
DCT(0.5) & MesoNet	DeepFakeFace	52.1 ± 1.0	78.2 ± 10.6	5.8 ± 2.3
	FaceForensics++	68.4 ± 1.2	68.1 ± 2.5	69.6 ± 4.5
	Individual	62.8 ± 3.0	61.7 ± 3.6	68.0 ± 5.9
	ProGAN	98.2 ± 2.6	96.8 ± 4.6	100.0 ± 0.0
	StarGAN	88.3 ± 0.7	86.8 ± 1.6	90.4 ± 1.6
	WhichFace	93.6 ± 1.2	89.6 ± 1.4	98.6 ± 0.8
	Overall	**72.7 ± 0.7**	**75.5 ± 2.3**	**67.2 ± 2.6**
Dolos & MesoNet	DeepFakeFace	55.9 ± 6.5	48.4 ± 27.1	34.3 ± 32.8
	FaceForensics++	81.3 ± 1.4	89.1 ± 0.8	71.2 ± 2.7
	Individual	60.2 ± 2.5	58.0 ± 2.2	73.6 ± 2.2
	ProGAN	58.2 ± 3.7	77.3 ± 13.1	24.0 ± 4.2
	StarGAN	90.3 ± 0.7	88.6 ± 0.7	92.5 ± 1.7
	WhichFace	60.2 ± 2.2	59.8 ± 2.8	63.3 ± 4.9
	Overall	**72.6 ± 0.8**	**74.8 ± 1.5**	**68.2 ± 4.7**

References

[1] K. Conger. *Hackers' Fake Claims of Ukrainian Surrender Aren't Fooling Anyone. So What's Their Goal?* 2022. URL: https://www.nytimes.com/2022/04/05/us/politics/ukraine-russia-hackers.html.

[2] A. Satariano and P. Mozur. *The People Onscreen Are Fake. The Disinformation Is Real.* 2023. URL: https://www.nytimes.com/2023/02/07/technology/artificial-intelligence-training-deepfake.html.

[3] H. Chen and K. Magramo. *Finance worker pays out $25 million after video call with deepfake 'chief financial officer'*. 2024. URL: https://www.cnn.com/2024/02/04/asia/deepfake-cfo-scam-hong-kong-intl-hnk/index.html.

[4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. "Generative adversarial nets". In: *Advances in neural information processing systems* 27 (2014).

[5] T. Karras, S. Laine, and T. Aila. "A style-based generator architecture for generative adversarial networks". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 4401–4410.

[6] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. "Analyzing and Improving the Image Quality of StyleGAN". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020.

[7] D. P. Kingma, M. Welling, et al. "An introduction to variational autoencoders". In: *Foundations and Trends in Machine Learning* 12.4 (2019), pp. 307–392. DOI: 10.1561/2200000056.

[8] M. Masood, M. Nawaz, K. M. Malik, A. Javed, A. Irtaza, and H. Malik. "Deepfakes generation and detection: State-of-the-art, open challenges, countermeasures, and way forward". In: *Applied intelligence* 53.4 (2023), pp. 3974–4026.

[9] J. Ho, A. Jain, and P. Abbeel. "Denoising diffusion probabilistic models". In: *Advances in neural information processing systems* 33 (2020), pp. 6840–6851.

[10] P. Dhariwal and A. Nichol. "Diffusion models beat gans on image synthesis". In: *Advances in neural information processing systems* 34 (2021), pp. 8780–8794.

[11] A. Lugmayr, M. Danelljan, A. Romero, F. Yu, R. Timofte, and L. Van Gool. "Repaint: Inpainting using denoising diffusion probabilistic models". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, pp. 11461–11471.

[12] T. Zhao, X. Xu, M. Xu, H. Ding, Y. Xiong, and W. Xia. "Learning Self-Consistency for Deepfake Detection". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 15023–15033.

[13] X. Li, K. Yu, S. Ji, Y. Wang, C. Wu, and H. Xue. "Fighting against deepfake: Patch&pair convolutional neural networks (PPCNN)". In: *Companion Proceedings of the Web Conference 2020*. 2020, pp. 88–89.

[14] Y.-J. Heo, W.-H. Yeo, and B.-G. Kim. "Deepfake detection algorithm based on improved vision transformer". In: *Applied Intelligence* 53.7 (2023), pp. 7512–7527.

[15] M. A. Arshed, S. Mumtaz, M. Ibrahim, C. Dewi, M. Tanveer, and S. Ahmed. "Multi-class ai-generated deepfake face detection using patch-wise deep learning model". In: *Computers* 13.1 (2024), p. 31.

[16] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros. "CNN-generated images are surprisingly easy to spot... for now". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 8695–8704.

[17] A. Malik, M. Kuribayashi, S. M. Abdullahi, and A. N. Khan. "DeepFake Detection for Human Face Images and Videos: A Survey". In: *IEEE Access* 10 (2022), pp. 18757–18775. DOI: 10.1109/ACCESS.2022.3151186.

[18] C.-C. Hsu, Y.-X. Zhuang, and C.-Y. Lee. "Deep fake image detection based on pairwise learning". In: *Applied Sciences* 10.1 (2020), p. 370.

[19] S. Fung, X. Lu, C. Zhang, and C.-T. Li. "Deepfakeucl: Deepfake detection via unsupervised contrastive learning". In: *2021 international joint conference on neural networks (IJCNN)*. IEEE. 2021, pp. 1–8.

[20] S. Dong, J. Wang, R. Ji, J. Liang, H. Fan, and Z. Ge. "Implicit identity leakage: The stumbling block to improving deepfake detection generalization". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2023, pp. 3994–4004.

[21] D.-C. Stanciu and B. Ionescu. "Improving generalization in deepfake detection via augmentation with recurrent adversarial attacks". In: *Proceedings of the 3rd ACM International Workshop on Multimedia AI against Disinformation*. 2024, pp. 46–54.

[22] P. Korshunov and S. Marcel. "Improving generalization of deepfake detection with data farming and few-shot learning". In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 4.3 (2022), pp. 386–397.

[23] Y. Bian and H. Chen. "When does diversity help generalization in classification ensembles?" In: *IEEE Transactions on Cybernetics* 52.9 (2021), pp. 9059–9075.

[24] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler. "The power of ensembles for active learning in image classification". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 9368–9377.

[25] A. Kumar, J. Kim, D. Lyndon, M. Fulham, and D. Feng. "An ensemble of fine-tuned convolutional neural networks for medical image classification". In: *IEEE journal of biomedical and health informatics* 21.1 (2016), pp. 31–40.

[26] Á. Casado-García and J. Heras. "Ensemble methods for object detection". In: *ECAI 2020*. IOS Press, 2020, pp. 2688–2695.

[27] L. Rokach. "Ensemble-based classifiers". In: *Artificial intelligence review* 33 (2010), pp. 1–39.

[28] H. Song, S. Huang, Y. Dong, and W.-W. Tu. *Robustness and Generalizability of Deepfake Detection: A Study with Diffusion Models*. 2023. arXiv: 2309.02218 [cs.CV].

[29] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner. "FaceForensics++: Learning to Detect Manipulated Facial Images". In: *International Conference on Computer Vision (ICCV)*. 2019.

[30] M. Rahman. *Individualized Deepfake Detection Dataset*. 2024. DOI: 10.21227/w7ma-fp34. URL: https://dx.doi.org/10.21227/w7ma-fp34.

[31] *faceswap-GAN*. 2022. URL: https://github.com/shaoanlu/faceswap-GAN.

[32] T. Karras, T. Aila, S. Laine, and J. Lehtinen. "Progressive growing of gans for improved quality, stability, and variation". In: *arXiv preprint arXiv:1710.10196* (2017).

[33] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo. "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018, pp. 8789–8797.

[34] J. West and C. Bergstrom. *Which Face Is Real?* 2019. URL: https://whichfaceisreal.com/.

[35] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen. "Mesonet: a compact facial video forgery detection network". In: *2018 IEEE international workshop on information forensics and security (WIFS)*. IEEE. 2018, pp. 1–7. DOI: 10.1109/WIFS.2018.8630761.

[36] J. Ricker, S. Damm, T. Holz, and A. Fischer. "Towards the detection of diffusion model deepfakes". In: *arXiv preprint arXiv:2210.14571* (2022). DOI: 10.48550/arXiv.2210.14571.

[37] D.-C. Țânțaru, E. Oneață, and D. Oneață. "Weakly-supervised deepfake localization in diffusion-generated images". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2024, pp. 6258–6268. DOI: 10.48550/arXiv.2311.04584.

[38] L. Chai, D. Bau, S.-N. Lim, and P. Isola. "What makes fake images detectable? understanding properties that generalize". In: *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*. Springer. 2020, pp. 103–120. DOI: 10.1007/978-3-030-58574-7_7.

[39] F. Chollet. "Xception: Deep learning with depthwise separable convolutions". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1251–1258.

[40] A. Minz and C. Mahobiya. "MR image classification using adaboost for brain tumor type". In: *2017 IEEE 7th international advance computing conference (IACC)*. IEEE. 2017, pp. 701–705.

[41] J. Cao, L. Chen, M. Wang, H. Shi, and Y. Tian. “A parallel Adaboost-backpropagation neural network for massive image dataset classification”. In: *Scientific reports* 6.1 (2016), p. 38201.