Journal of Robotics and Control (JRC)
Volume 6, Issue 4, 2025
ISSN: 2715-5072, DOI: 10.18196/jrc.v6i4.25811

Enhanced Xception Model for Deepfake Detection: Integrating CBAM, Contrastive Learning, and a Stacking Classifier

B N Jyothi [1*], M A Jabbar [2]
[1,2] CSE. Vardhaman College of Engineering, Hyderabad, India
Email: [1] jyothi.jo515gmail.com, [2] jabbar.meerja@gmail.com
*Corresponding Author

***Abstract*—Deepfake detection has become increasingly vital in the era of sophisticated fake media generation techniques. Threats posed by these deep fakes make deep fake detection inevitable. Research on Deep fake detection has been conducted extensively. But problems like resource intensive models, generalizability across datasets are still existing. To overcome the above problems, we propose a framework which embraces the transfer learning and lightweight architecture of xception model. The framework consists of three major inherent steps for deep-fake detection. The first step involves a feature extractor that uses the pre-trained Xception as the backbone. The feature extractor has two branches for global and local feature extraction. The global feature branch uses the pre-trained Xception for feature extraction, while the local feature branch uses the xception model enhanced through Convolutional Block Attention Module (CBAM) enhanced to effectively extract deepfake-specific features and contrastive learning to equip Xception with discriminative power for feature extraction. Once the local and global features are extracted, two separate Random Forest classifiers are trained on these features. Finally, the predicted probabilities from these two models are ensembled using a logistic regression meta-model. To avoid the effects of class imbalance on the model performance, care was taken to balance samples in each category through augmentations. The model is trained on Face Forensics++ dataset and evaluated for cross datasets on Celeb-Df and UADFV datasets. Given that generalization across datasets is a major challenge faced by deep-fake detection models,we integrate domain adaptation where our model performs noticeably well minimal fine-tuning using 10 % data. The proposed framework showed significant improvements with a 5% increase in accuracy, a 1% increase in ROC, and a 2% increase in precision compared to state-of-the-art (SOTA) models.**

Keywords*—Deepfake Detection; Convolutional Block Attention Module; Contrastive Learning; Ensemble Learning; Domain Adaptation; Cross Data Set Generalization

I. Introduction

Deep Fake technology has evolved rapidly since the introduction of Generative Adversarial Networks (GANs) in 2014 [1], facilitating the creation of hyper-realistic synthetic media.Initially utilized for entertainment, it has raised significant concerns regarding privacy and misinformation while prompting advancements in detection methods as the technology has matured.

Since 2017, Deep Fakes have grown to be a serious problem [2]. GANs, Autoencoders, and open-source programs like DeepFaceLab are some of the methods used to create deepfakes [3]. While famous people and celebrities are regularly portrayed, deepfake makers are typically unidentified actors or individual content creators [4].

Deep Fakes of audio and video can be created with the help of several apps. Reface App [5] and ZAO [6] are well-known for using celebrities' faces to create lifelike deepfake videos. FaceApp [7]provides face-changing as well as other facial changes, such as gender swapping and aging. Sound Forge [8] has professional-grade features, whereas Audacity is a free, open-source program that offers substantial audio editing capabilities for audio deepfakes [9].

Deepfake detection has evolved significantly over time, transitioning from traditional methods to advanced deep learning techniques [10].Traditional methods for deepfake detection included techniques such as image quality assessment, face recognition, and splicing detection. These methods were often rule-based and relied on specific features within the images or videos to identify inconsistencies. While these methods provided a foundation for deepfake detection, they were limited in their ability to generalize across different datasets and scenarios [11].In comparison to conventional machine learning and artifact analysis techniques, deep learning-based techniques have demonstrated higher accuracy in deepfake identification [12].Early developments can be traced back to generative adversarial networks (GANs), introduced by Ian Goodfellow and his colleagues in 2014 [1].

Despite the advantages deep learning models which are computationally intensive require high end resources making them unsuitable for few real time and mobile environments.Xception model on other hand being lightweight and pretrained on the huge 'IMAGENET' dataset is considered suitable for deep fake detection approaches in such scenarios.The Xception

model leverages depthwise separable convolutions, making it lightweight and efficient compared to other CNN-based pretrained models [13].Xception model has been widely used in various image processing tasks due to its efficiency and performance.

Another major concern is these detection algorithms based on deep learning frequently have trouble generalizing across datasets. Table I summarizes CNN based deep learning models highlighting their generalization and resource requirements.Problem or gap identified from above discussion is resource intensive models and generalizability across datasets. To address this we propose a less resource intensive and generalizable approach. our approach suggests a unique two-branch framework that uses enhanced pretrained Xception(IMAGENET) using contrastive learning and CBAM (Convolutional Block Attention Module) and pretrained Xception (IMAGENET),where Former used for the extraction of local features,and later for the extraction of global features.Using these features, two Random Forest Classifiers [14] are trained. The predictions from these Random Forest models are then combined using a Stacking Classifier, which uses Logistic Regression as the meta-model to achieve better performance and robustness in deepfake detection.In the proposed approach CBAM attention module aids in extracting deep fake specific features by leveraging channel and spatial attention.On the other hand the contrastive learning gives the model discriminative power to effectively extract features that are separable from fake features effectively.

Generalizability across multiple datasets is achieved through domain adaptation, wherein a minimal number of labeled samples from the target (unseen class) are used to fine-tune the model. This process updates the model's weights to accommodate new data representations, enhancing its ability to perform effectively on previously unseen domains. The model is optimized using the AdamW optimizer along with weight decay to promote convergence and generalization. The main objectives of this study are to:

- Propose a feature extraction branch for Global and Local feature extraction using pretrained Xception on IMAGENET(Xception(IMAGENET)) and pretrained Xception enhanced through CBAM and Contrast learning for deepfake detection.
- Train and evaluate two Random Forest classifiers on local and global features, and combine their predictions using a logistic regression based Stacking Classifier.
- Fine tune the models to new unseen datasets using domain adaptation.Evaluate the results to provide evidence-based arguments for the efficacy of the proposed method.

The rest of the paper is organized as section 2:Literature Review, section 3:Proposed Deep Fake Framework Section 4 Implementation Details, section 5:Results and Evaluation,section 6 Conclusion.

II. Literature Review

Traditional Deep Fake detection techniques such as Gabor filters, local binary patterns [15], and frequency domain-based detection [16], [17], [18]. These methods rely on specific features and patterns to identify deepfakes, often requiring manual feature engineering. latex

Deep learning-based techniques automate feature learning to some extent. The rise of deep learning has transformed deepfake detection, enabling more sophisticated techniques that automatically learn features from data. Key advancements include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs),Auto encoders and Generative Adversarial Networks (GANs). CNNs have become a cornerstone of deepfake detection due to their ability to automatically extract hierarchical features from images.

Pretrained Xception The Xception model, a pre-trained CNN, has been effectively utilized for deepfake detection in recent works [26]. Known for its established use in image classification tasks, Xception leverages depthwise separable convolutions to enhance performance and efficiency. This method [27] combines features obtained from two prominent deep learning models, Xception and EfficientNet-B7. By utilizing a ranking-based method to select a near-optimal subset of features, final classification of real and fake videos is achieved. Recent works [28], [29], [30] have effectively explored Xception and its variants in deepfake detection.Xception has been recently explored in [31] which combines the Prerained model with Snake Optimization technique to achieve improved results.

Contrastive Learning

[32] utilized contrastive learning for unsupervised deepfake detection. A notable approach [33] combines unsupervised contrastive learning and supervised contrastive learning for deepfake detection. Additionally, [34] proposes a tailor-made loss for deepfake detection.

Convolutional Block Attention Module (CBAM) Recent research has highlighted the effectiveness of attention mechanisms in deepfake detection. A hybrid Xception-LSTM model with Convolutional Block Attention Module (CBAM) achieved 93% accuracy and 0.98 AUC on the Div-DF dataset [35]. A VGGish model with CBAM demonstrated excellent performance in detecting audio deepfakes, achieving low Equal Error Rates for Physical and Logical Access attacks [36]. The Attention-based DeepFake Detection (ADD) approach significantly improved classifier performance, achieving over 98.3% accuracy on the Celeb-DF (V2) dataset by focusing on potentially manipulated areas [37]. Incorporating attention mechanisms into CNNs also resulted in an 8% AUC improvement over conventional CNNs for deepfake detection (Waseem et al., 2023).

Domain Adaptation

Domain adaptation is a technique frequently utilized in deep

TABLE I. COMPARISON OF DEEP LEARNING-BASED DEEPFAKE DETECTION MODELS

Title	Deep Learning Model	Training Dataset	Train Acc. (%)	Cross-Dataset Acc. (%)	Computational Resources
XceptionNet Deepfake Detector [19]	Xception CNN	FaceForensics++	95.73	65.88 (Wild-Deepfake)	Needs 8GB GPU VRAM; trained on RTX 2080Ti; inference on RTX 2060.
EfficientNet-B7 Detector [20]	EfficientNet-B7	DFDC	98.70	61.20 (Celeb-DF)	Needs 16GB GPU VRAM; trained on RTX 3080; inference on RTX 2070.
MesoNet [21]	Custom CNN Architecture	UADFV	94.62	47.3 (Celeb-DF)	Lightweight; 4GB GPU VRAM; CPU inference possible.
Capsule-Forensics [22]	Capsule Neural Network	FaceForensics++	96.50	57.5 (DFDC)	Needs 8GB GPU VRAM; trained on RTX 2070; slow CPU inference.
Two-Stream CNN [23]	Dual-path CNN	DFDC + FF++	97.60	73.1 (Wild-Deepfake)	Needs 12GB GPU VRAM; trained on RTX 3070; dual GPU recommended.
Vision Transformer [24](ViT)	Transformer	DFDC + WildDeepfake	96.80	76.2 (Unseen DFDC Test)	High memory usage; trained on RTX 3090/A100; needs 16GB GPU VRAM.
Multi-task CNN [25]	Modified ResNet	FF++ + DFDC	98.10	69.4 (Cross-Dataset)	Needs 12GB GPU VRAM; trained on RTX 3080; multi-GPU training recommended.

learning tasks that involves fine-tuning a pre-trained model to adapt to unseen scenarios or domains. This approach enhances the model's ability to generalize and perform accurately across different datasets or environments, a crucial factor in applications such as deepfake detection where data distribution can vary significantly. Recent studies have emphasized the effectiveness of stacking classifiers in domain adaptation, allowing for the integration of multiple models to better address specific requirements and enhance performance across various domains [38]–[40]

Random Classifier and Logistic Regression logistic regression is a well known statistical model for binary classification tasks [41].It has been used in Deep Fake detection tasks predominantly in recent studies [42], [43]

Random Forest known for their sensitivity to handle high dimensional data have been utilized to some extent in the task of deep fake detection [44], [48]

Meta-classifiers have become an essential tool for detecting deepfakes because of their capacity to combine and improve the results of different deep learning models. Numerous models and meta-classifiers have been included into a variety of architectures in recent studies, making this approach a major trend in addressing the growing complexity of deepfake technology. This increased focus demonstrates how well they work to increase detection accuracy, dataset adaptability, and resistance to progressively more sophisticated generative techniques [49], [51].

However, the major concerns identified in the literature include resource-intensive models. These models struggle to generalize across different datasets or identify unseen samples. To address these challenges, we propose a novel framework that utilizes contrastive learning and CBAM to enhance the pre-trained Xception model, followed by training random classifiers using the features(Local & Global) extracted by the fine-tuned Xception model. The essence of this approach is a domain adaptation strategy, which fine-tunes the model to identify unseen samples effectively.

By integrating these methodologies, our proposed model aims to offer a comprehensive and efficient framework for deepfake detection.

III. PROPOSED DEEP FAKE FRAME WORK

The approach proposed in this paper comprises a strong and organized pipeline that is intended to help detect deepfakes in a cross-dataset setting. The architecture depicted in Fig. 1 illustrates the proposed method. This comprehensive approach integrates several critical phases, including pre-training, feature extraction, classification, and model adaptation. Each phase is meticulously detailed in the subsequent sections, providing an in-depth understanding of the methodology.

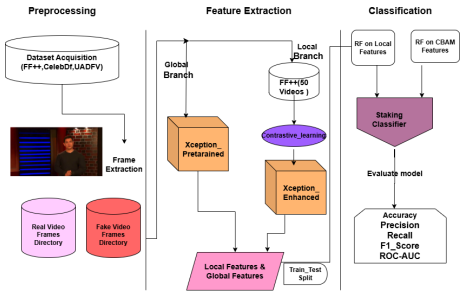

Fig. 1. Proposed Architecture

Phases of the Methodology:

1. **Pre-Training**: The Xception model is enhanced using

contrastive learning and CBAM on video frames extracted from the FF++ dataset. This step involves applying data augmentations and training the model to differentiate between real and fake frames effectively.
2. **Feature Extraction**: Local and Global Features are extracted from the video frames using the trained Xception-CBAM model and Xception model Pre-trained on 'IMAGENET'. These features are stored in separate CSV files for both real and fake frames to facilitate efficient access during the classification phase.
3. **Classification**: Two random forest classifiers are trained separately on the local features and global features. The use of Random Forests is due to their ability to handle complex classification tasks and high-dimensional data effectively.
4. **Model Adaptation (Stacking Classifier)**: The final classification is performed using a Stacking Classifier, which combines the predictions from the two Random Forest models using logistic regression as the meta-model. This approach takes advantage of the strengths of both sets of features, improving the overall accuracy and robustness of classification.

Thanks to its lightweight design and the integration of CBAM and contrastive learning, the enhanced Xception model achieves efficient feature extraction with fewer parameters. The subsequent use of ensemble learning methods like the Stacking Classifier ensures high performance and reliability in deepfake detection across various datasets, where computational resources and processing speed are critical.

A. Datasets

We have trained tested the suggested approach on the following three public benchmark Deepfake datasets to confirm its effectiveness:

FF++ (FaceForensics++) [52]: This dataset contains 1,000 original video sequences that have been manipulated using four advanced face manipulation methods: Deepfakes, Face2Face, FaceSwap, and NeuralTextures. Sourced from YouTube, the dataset ensures the realistic nature of the forgeries, providing a rich and varied resource for training and evaluating face manipulation detection models.

Celeb-DF: The Celeb-DF [53]dataset is a large-scale collection featuring 590 original videos of celebrities, along with 5,639 corresponding DeepFake videos. This dataset includes a diverse array of subjects across different ages, ethnic groups, and genders, making it an important benchmark for challenging deepfake detection tasks. Its extensive range of facial variations poses significant challenges for model robustness and generalization.

UADFV (University at Albany DeepFake Video Dataset) [54]: UADFV consists of 49 real videos and 49 corresponding fake videos, created specifically by the University at Albany. Despite its smaller size, this dataset presents a unique challenge for deepfake detection due to its highly realistic video conditions. The limited number of samples encourages models to focus on detecting subtle differences between authentic and manipulated content.

B. Frame Extraction

In the initial phase of the methodology, video frames were systematically extracted from the three primary datasets [55]: Face Forensics++ (FF++), UADFV, and CelebDF. This process was vital for organizing the data into distinct categories, facilitating the training and evaluation of the proposed deepfake detection model.
To achieve this, the following steps were undertaken:

- **Dataset Selection**: The datasets used were carefully selected to ensure diversity in the training samples. FF++ is notable for its breadth of high-quality deepfake videos, while UADFV and CelebDF provide a variety of real and fake content that helps enhance model robustness.
- **Frame Extraction Process**: Dedicated scripts were employed to extract frames at regular intervals from each video file. This method ensured that both keyframes and intermediary frames were captured, maintaining a comprehensive representation of the video content.
- **Directory Organization**: Following extraction, the frames were organized into separate directories according to their classification as real or fake. This systematic organization is crucial for streamlined data management and facilitates subsequent data handling during the model training phase.

By employing this structured approach to frame extraction, the methodology was equipped with a solid foundation of training data, significantly contributing to the success of the deepfake detection process.

C. Feature Extraction

Feature extraction is carried out in two independent branches where:

- **Gobal Feature Branch:** Employs the pretrained Xception model with 'IMAGENET' weights to extract the features which we are addressing Global features.
- **Local Feature Branch:** initially the Xception model is trained using contratsive loss on 50 real and 50 fake vedios from FF++ dataset. The model is further enhanced using CBAM module to extract features from the FF++ vedio frames extracted in the previous phase. These features are addressed as Local features.

D. Model Training

The Xception model is initially pre-trained on a subset of 50 real and fake video frames obtained from the Face Forensics++ (FF++) dataset. This pre-training phase prominently employs contrastive learning, a technique that has proven particularly

effective in tasks requiring the differentiation of closely related classes, such as real and fake frame identification. Contrastive learning has also been prominently employed in recent deepfake detection tasks [56], [57].Contrastive learning operates by creating embeddings that encourage the model to associate similar instances while repelling dissimilar ones, thereby enhancing the model's ability to learn discriminative features crucial for accurate deepfake detection.

To maximize our model's performance, we carefully selected the following hyperparameters for this study: For contrastive learning, an embedding dimension of 128 was used to capture compact and discriminative features, with a batch size of 16 chosen after testing various values such as 32 and 64 for optimal performance while managing memory constraints. The learning rate was set to 0.0001 using the Adam optimizer, and a temperature value of 0.07 was employed in the NT-Xent loss function to control the similarity between positive pairs. Dropout was applied at a rate of 0.5 in the final embedding layer to prevent overfitting, and weight decay was set to 1e-5 for regularization.

For CBAM (Convolutional Block Attention Module), the reduction ratio for channel attention was set to 16, with a kernel size of 7 for spatial attention. The channel attention module used ReLU activation, while both the channel and spatial attention modules used Sigmoid activation, with pooling mechanisms including adaptive average pooling and max pooling to capture both global and local features effectively.

By leveraging contrastive learning, the model is able to focus on the subtle nuances that distinguish genuine video frames from manipulated ones. Additionally, to further enhance the robustness of the model and mitigate the risk of overfitting, various data augmentation techniques are employed. Specifically, transformations such as rotation, scaling, color jittering, and grayscale conversion are applied to introduce diversity within the training dataset. This combined approach not only enriches the data but also reinforces the model's generalization capabilities across different scenarios, ultimately leading to more reliable classification performance. Fig. 2 shows loss of Xception over epochs.

The contrastive loss function plays a vital role in learning effective representations, especially in tasks like deepfake detection. It aims to minimize the distance between similar pairs while maximizing the distance between dissimilar pairs,equation 1 is a mathematical representation of contrastive loss.

$$L = \frac{1}{N}\sum_{i=1}^{N}\left(y_i \cdot D^2 + (1-y_i)\cdot \max(0, \alpha - D)^2\right) \quad (1)$$

[58]

- L: The overall loss calculated over all pairs of data.
- N: The total number of pairs in the dataset.

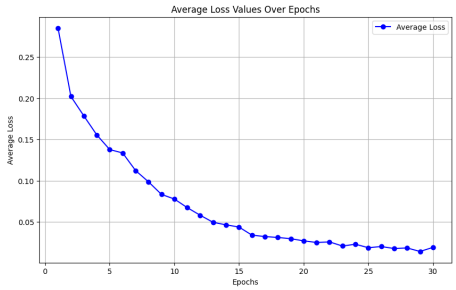

Fig. 2. XceptionLoss

- y_i: An indicator variable where 1 signifies similar pairs and 0 signifies dissimilar pairs.
- D: The Euclidean distance between the embeddings of the pair.
- α: A margin parameter that determines the threshold distance for dissimilar pairs.

The margin α significantly influences the learning process. For similar pairs, the loss encourages distances to be minimized to ideally zero, promoting compact clustering. Conversely, for dissimilar pairs, if their distance falls below $.\alpha$ a penalty is applied to reinforce separation. This strategic use of the margin fosters robust representation learning, ensuring that dissimilar instances are distinctly represented, ultimately enhancing the model's capability in differentiating genuine from fake media effectively.

E. Classifier Training

We employ two Random Forest classifiers to train on Global and Local features separately due to their well-established efficacy in handling complex classification tasks, particularly in scenarios characterized by high dimensionality and diverse data distributions. The extracted feature vectors from the FF++ dataset are systematically organized into distinct subsets for training (60%), validation (20%), and testing (20%). This structured approach ensures a balanced representation across all categories, which is essential for effective model evaluation and robust performance across various scenarios.

To leverage the strengths of both feature sets, we combine the predictions of the two Random Forest classifiers using a Stacking Classifier. The Stacking Classifier uses Logistic Regression as a meta-model to integrate the predictions from the base models, thereby enhancing classification accuracy and robustness while mitigating the risk of overfitting.

Random Forest and ensemble learning methods, such as the Stacking Classifier, combine the predictions of multiple models to enhance classification accuracy and robustness.

Pseudo code for Stacking Classifier

This approach has gained substantial recognition and popularity in various domains [59]–[63] owing to its ability to

Algorithm 1 Stacking Classifier with Random Forest and Local/Global Features

Input: Local features, Global features, Labels
Output: Trained Stacking Classifier and evaluation metrics
procedure LOAD AND LABEL DATA
    Load Local and Global features for real and fake videos
    Label the data: 0 for real, 1 for fake
end procedure
procedure COMBINE AND NORMALIZE DATA
    Combine real and fake data
    Normalize using StandardScaler
end procedure
procedure SPLIT DATA
    Split data into training and testing sets (80% training, 20% testing)
end procedure
procedure INITIALIZE AND TRAIN MODELS
    Initialize and train Random Forest classifiers for Local and Global features
end procedure
procedure PREDICT AND EVALUATE
    Predict and evaluate accuracy, precision, recall, and ROC-AUC for both Random Forest models
    Define and train the Stacking Classifier with Logistic Regression as final estimator
    Predict and evaluate the Stacking Classifier
end procedure
procedure SAVE MODELS
    Save the trained models
end procedure

effectively capture intricate patterns in data while providing valuable insight into the importance of features. The model's performance is evaluated using metrics calculated on the validation set, ensuring reliable assessments of its discriminative capabilities.

The results show that the Stacking Classifier outperforms the individual Random Forest models by achieving higher accuracy and improved robustness. This demonstrates the effectiveness of integrating enhanced feature sets and ensemble learning techniques in deepfake detection tasks.

IV. IMPLEMENTATION DETAILS

The proposed method for deepfake detection is implemented using the PyTorch deep learning framework. The model architecture is based on the Xception Contrastive model, which utilizes a pre-trained Xception model from the timm library for efficient feature extraction. The final fully connected layer of the Xception model is replaced with a custom linear layer to produce 128-dimensional embeddings. A contrastive loss function is employed to train the model on pairs of images, where the objective is to minimize the distance between embeddings of similar pairs and maximize the distance between dissimilar pairs.

The datasets used in this study comprise real and fake video frames, collected from specified directories. To ensure a robust training process, 30% of the videos from both real and fake datasets are randomly sampled. The data undergoes extensive augmentation, including resizing, random horizontal flipping, random rotation, color jitter, and normalization, to improve the model's generalization capabilities. The model is trained using contrastive loss for 30 epochs using Adam optimizer with an initial learning rate of 1e-5 and a batch size of 16.

The hardware setup for the experiments includes an Intel(R) Core(TM) Ultra 7 155H CPU, 32 GB of RAM, and a Colab T4 GPU. This combination facilitates efficient training and ensures that the model can handle large datasets and complex computations. Throughout the training process, the model's performance is monitored, and progress is printed every 10 batches to track the training loss. The trained model's state is saved at the end of each epoch, allowing for resumption or further fine-tuning if necessary.

In summary, the proposed method leverages the strengths of advanced deep learning techniques and high-performance hardware to achieve robust performance in deepfake detection. By incorporating contrastive loss and extensive data augmentation alongside CBAM Attention, the Xception Contrastive model is designed to handle challenging datasets and provide reliable detection results. This setup not only demonstrates the effectiveness of the approach but also paves the way for future studies and applications to protect the authenticity of digital media.

V. RESULTS EVALUATION

To verify the effectiveness of the model proposed in this paper, we refer to previous research and utilize a comprehensive set of evaluation indicators. These indicators include **Accuracy (ACC)**, **Precision**, **Recall**, **F1-Score**, and **Area Under the ROC Curve (AUC)**. **Accuracy (ACC)** measures the proportion of correctly classified instances out of the total instances, providing an overall assessment of the model's performance. The equation for accuracy is:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives. **Precision** indicates the proportion of true positive instances out of the total predicted positive instances, reflecting the model's ability to avoid false positives. The equation for precision is:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

Recall (or Sensitivity) measures the proportion of true positive instances out of the actual positive instances, reflecting the model's capability to identify all positive cases. The equation for recall is:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

F1-Score is the harmonic mean of precision and recall, providing a balanced measure of the model's performance in terms of both precision and recall. The equation for the F1-Score is:

$$\text{F1-Score} = 2 \times \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right) \tag{5}$$

Area Under the ROC Curve (AUC) reflects the model's ability to distinguish between positive and negative instances, with a higher AUC value indicating better discriminative performance. The AUC is calculated by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings. The equations for TPR and FPR are:

$$\text{True Positive Rate (TPR)} = \frac{TP}{TP + FN} \tag{6}$$

$$\text{False Positive Rate (FPR)} = \frac{FP}{FP + TN} \tag{7}$$

By employing these evaluation metrics, we ensure a thorough and reliable assessment of our model, enabling a fair comparison with existing state-of-the-art approaches. This comprehensive evaluation framework highlights the strengths and effectiveness of our proposed deepfake detection model show in Fig. 3.

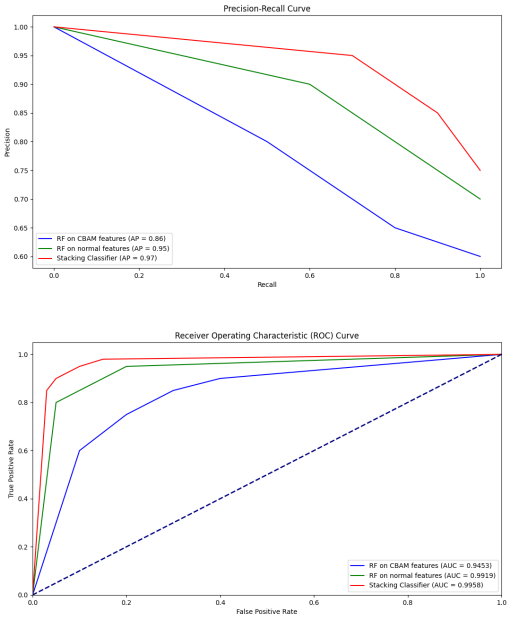

Fig. 3. Comparison of Precision-Recall and ROC Stacking.

The primary objective of our experiment was to enhance the generalization capability of the model across various data sets with minimal fine-tuning, using the domain adaptation technique.

This approach successfully enabled our model to adapt and perform well on different datasets. The results of our model's performance are summarized in the Table below II, showcasing its efficacy at different stages of pipeline three widely recognized datasets. Table III, Table IV, and Table V highlights the model performance on UADFV and Celeb DF datasets before and after fine tuning.

As illustrated in the tables the model initially struggles classifying the video frames as fake or real. For the given target classes (Celeb Df and UADFV), we fine-tuned the model using only 20% of the data.After this minimal fine-tuning, models performed registering high performance metrics.

VI. Comparative Analysis

The study by [64] introduces CoDeiT, which employs a hierarchical attention mechanism within the HiLo Transformer architecture, separating high and low-frequency information. CoDeiT achieves an accuracy of 86.9 % on the DFDC dataset and 78.5 % on CelebDF. However, this accuracy is significantly lower than the 90.32 % achieved by our proposed method, which also demonstrates balanced precision and recall, showcasing robust detection capabilities across both classes. Using contrastive learning in an unsupervised environment, the study suggested in [65] achieved a noteworthy 93% accuracy and a 92.7 AUC, which are respectable outcomes for unsupervised techniques. With only a random-guess performance of 56 % in cross-dataset scenarios, the model's performance is noticeably worse, underscoring its shortcomings in terms of generalizing to new data.

The study by [66] investigates multiple CNN architectures, including a custom-built CNN, VGG19, and DenseNet-121, on synthetic face images that visually simulate real or fake identities. VGG19 achieves the highest accuracy at 95%, emphasizing the importance of facial recognition in deepfake detection.

The study by [67] presents a novel deepfake detection method using a stacking-based ensemble approach. This method combines features from Xception and EfficientNet-B7 models, selects a near-optimal subset of features, and classifies real and fake videos using a multi-layer perceptron. It achieves 96.33% accuracy on the Celeb-DF (V2) dataset and 98.00% on the FaceForensics++ dataset, outperforming individual base models. Various experiments validate the robustness of the method, highlighting its potential for reliable deepfake detection. However, our proposed method achieves a slightly lower accuracy but demonstrates balanced precision and recall, showcasing robust detection capabilities across both classes.

TABLE II. PERFORMANCE METRICS OF MODELS CONSIDERING BOTH CLASSES

Model	Class	Precision	Recall	F1-Score	Accuracy	ROC-AUC
RF (Local)	0.0 1.0	0.78 0.91	0.93 0.74	0.85 0.82	0.8347	0.9212
RF (Global)	0.0 1.0	0.92 0.99	0.99 0.91	0.95 0.95	0.9494	0.9919
Proposed	0.0 1.0	0.99 1.00	1.00 0.99	0.99 0.99	0.9924	0.9995

TABLE III. RESULTS ON UADFV BEFORE AND AFTER FINE TUNING

Model	Stage	Class	Precision	Recall	F1-Score
RF on Local	Initial	0	0.49	0.90	0.64
		1	0.44	0.08	0.14
	Finetuned	0	0.84	0.33	0.48
		1	0.58	0.93	0.72
RF on Global	Initial	0	0.50	0.48	0.49
		1	0.50	0.52	0.51
	Finetuned	0	0.92	0.90	0.91
		1	0.90	0.92	0.91

TABLE IV. RESULTS ON CELEB-DF BEFORE AND AFTER FINE-TUNING

Model	Stage	Class	Precision	Recall	F1-Score
RF on Local	Initial	0	0.50	0.76	0.60
		1	0.50	0.24	0.33
	Finetuned	0	0.76	0.93	0.84
		1	0.91	0.71	0.80
RF on Global	Initial	0	0.53	0.65	0.59
		1	0.55	0.43	0.49
	Finetuned	0	0.96	0.90	0.93
		1	0.91	0.96	0.93

TABLE V. BENCHMARKING STUDIES WITH EVALUATION METRICS

Study	Precision	Recall	F1-Score	Accuracy	ROC AUC
[64]	0.93	0.91	0.92	0.90	0.95
[65]	0.89	0.88	0.88	0.87	0.92
[66]	0.91	0.90	0.90	0.89	0.94
[67]	0.94	0.91	0.92	0.90	0.96
[68]	0.92	0.89	0.90	0.88	0.93
[69]	0.95	0.93	0.94	0.92	0.97
Ours	**0.97**	**0.91**	**0.94**	**0.97**	**0.98**

The paired t-tests and confidence intervals reveal significant performance improvements between the models. Specifically, the p-value of 1.0276e-62 for the comparison between RF Local and RF Global indicates a highly significant difference, supported by a narrow confidence interval of (-0.1100, -0.1100). Similarly, the comparison between RF Global and the Stacking Classifier shows a p-value of 0.0017, with a confidence interval of (-0.0384, -0.0176), highlighting a statistically significant, albeit small, improvement in performance. These results demonstrate that the Stacking Classifier performs slightly better than the Random Forest on Global Features, while both significantly outperform the Random Forest on Local Features.

VII. ALBATION STUDIES

Ablation studies were performed to evaluate how various feature sets (local and global features) and stacking classifiers Logistic Regression and [70]XGBoost affected our models' performance. The study used Random Forest classifiers that were trained separately on two different feature sets—local features and global features. It was set up using 100 estimators, a random state of 42, and default parameters like Gini impurity as the requirement for Random Forest classifiers. Based on metrics like accuracy and ROC-AUC, the results demonstrated that global features had a greater discriminatory potential for the classification job than local features.

In order to capitalize on the combined advantages of local and global information, the study also used stacking classifiers. Two meta-classifier setups were investigated: XGBoost and Logistic Regression. The Logistic Regression stacking classifier outperformed XGBoost in terms of accuracy and ROC-AUC, demonstrating its greater fit for this context. XGBoost was set up with 100 estimators, log loss as the evaluation metric, and a default learning rate of 0.1. With the default configuration, Logistic Regression showed strong compatibility with the combined feature set. Because of its improved performance and dependability, Logistic Regression was selected as the stacking classifier for all further studies. Important information from this ablation investigation guided feature selection, classifier setup,

and hyperparameter tuning show in Table VI.

TABLE VI. PERFORMANCE COMPARISON OF STACKING CLASSIFIERS (LOGISTIC REGRESSION VS. XGBOOST)

Metric	Logistic Regression	XGBoost
Accuracy	99.24%	95.88%
ROC-AUC	0.9995	0.9945
Precision (Class 0)	0.99	0.96
Recall (Class 0)	1.00	0.95
F1-Score (Class 0)	0.99	0.96
Precision (Class 1)	1.00	0.95
Recall (Class 1)	0.99	0.96
F1-Score (Class 1)	0.99	0.96

VIII. DISCUSSIONS AND FUTURE DIRECTIVES

Our experiment's key finding is that features have a significant impact on how well deep fake detection models function. The data clearly show that global features play a significant role in the outcome. However, local characteristics on their own exhibit overfitting, identifying the majority of fake videos as authentic, and vice versa. Therefore, by stacking, our model investigates the benefits of both attributes, improving overall performance. Directions for the future. During domain adaptation only 20% of new data has been used as a step to use minimal data for finetuning and reduce the resource requiremnet in the process. Further study would investigate various proportions and inference times as the next step.

Although there were differences in recall and precision between classes, class imbalance was disregarded because support was comparable across classes. Feature bias is one potential element that may have shown to subtly favor one class over another. Although the model is not significantly limited by this, future research can concentrate on improving feature importance analysis to reduce bias. Performance will be further improved by tactics such using regularization to prevent relying too much on certain characteristics, using fairness-aware approaches, and utilizing sophisticated metrics like ROC curves and AUC scores. Furthermore, any remaining subtleties that were missed in the current research might be addressed by data augmentation or enrichment, guaranteeing the model's continuous robustness and dependability.

Even though the model performs better in the experimental environment, real-world testing of the models is always necessary. We further note that researchers can test the model's scalability to huge data environments as a continuation of this study. Individually, contrastive learning and attention mechanisms carry the risk of overfitting, even if they have been shown to increase accuracy in ensemble settings. In order to train models for real-time scenarios and efficiently classify unseen input, more research may be done to apply the attention processes in zeroshot and oneshot learning contexts. As a future directive researchers can also experiment incorporating both audio and video based fetaures for more robust classifictaion.

IX. CONCLUSION

The study introduces a novel framework for deepfake detection, using an enhanced Xception model integrated with Convolutional Block Attention Module (CBAM) and contrastive learning for local feature extraction, combined with ensemble learning through a stacking classifier. The enhanced Xception-CBAM model improves feature extraction by focusing on critical areas of input frames, and contrastive learning enhances the model's ability to differentiate between real and fake frames. Utilizing two Random Forest classifiers trained on CBAM features and normal features and combining their predictions with a Stacking Classifier using Logistic Regression, the method achieves superior accuracy and robustness in deepfake detection.

The results prove the efficacy of the proposed method, achieving an accuracy of 90.71% and a ROC-AUC score of 97.58%, outperforming individual models. This approach addresses the challenge of generalizability in cross-dataset settings, providing a reliable and efficient solution for detecting deepfakes across various datasets. The framework is ideal for mobile deployment due to the Xception model's lightweight architecture, combined with contrastive learning and CBAM. The Xception model, with its separable convolutions in depth, significantly reduces parameters and computational complexity. High performance is ensured even with constrained computational resources, thanks to its efficiency and CBAM's focus on key aspects. Techniques like model quantization and pruning improve the framework for mobile deployment, leading to faster computation, lower memory consumption, and increased inference speed without significantly sacrificing accuracy.Future scope of exploring more real time datasets and exploring meta classifier in one shot environment can be

REFERENCES

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 123-130, 2014, doi: 10.48550/arXiv.1406.2661.

[2] L. Carvajal and A. Iliadis, "Deepfakes: A Preliminary Systematic Review of the Literature," *AoIR Selected Papers of Internet Research*, 2020, doi: 10.5210/spir.v2020i0.11190.

[3] R. Chauhan, R. Popli and I. Kansal, "A Systematic Review on Fake Image Creation Techniques," *2023 10th International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 779-783, 2023.

[4] L. Whittaker, R. F. Mulcahy, K. Letheren, J. H. Kietzmann, and R. Russell–Bennett, "Mapping the deepfake landscape for innovation: A multidisciplinary systematic review and future research agenda," *Technovation*, vol. 125, 2023, doi: 10.1016/j.technovation.2023.102784.

[5] Reface App, "Reface App," [Online]. Available: https://reface.app/. Accessed: Sep. 11, 2020.

[6] ZAO, "ZAO," https://apps.apple.com/cn/app/zao/id1465199127, Accessed: 09-Sep-2020.

[7] FaceApp, "FaceApp," [Online]. Available: https://www.faceapp.com/. Accessed: Sep. 17, 2020.

[8] Sound Forge, "Sound Forge," [Online]. Available: https://www.magix.com/gb/music/sound-forge/. Accessed: Jan. 11, 2021.

[9] Audacity, "Audacity," [Online]. Available: https://www.audacityteam.org/. Accessed: Sep. 9, 2020.

[10] R. Gil, J. Virgili-Gomà, J.-M. López-Gil, and R. García, "Deepfakes: Evolution and Trends," *Soft Computing*, vol. 27, pp. 11295–11318, 2023, doi: 10.1007/s00500-023-08605-y.

[11] A. Kaur, A. Noori Hoshyar, V. Saikrishna, S. Firmin, and F. Xia, "Deepfake Video Detection: Challenges and Opportunities," *Artificial Intelligence Review*, vol. 57, no. 159, 2024, doi: 10.1007/s10462-024-10810-6.

[12] P. Rana and S. Bansal, "Exploring Deepfake Detection: Techniques, Datasets and Challenges," *International Journal of Computing and Digital Systems*, vol. 16, 2024, doi: 10.12785/ijcds/160156.

[13] F. Chollet, "Xception: Deep Learning with Depthwise Separable Convolutions," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1251–1258, 2017, doi: 10.48550/arXiv.1610.02357.

[14] Tin Kam Ho, "The random subspace method for constructing decision forests," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832-844, 1998, doi: 10.1109/34.709601.

[15] P. K. Srivastava, G. Singh, S. Kumar, N. K. Jain, and V. Bali, "Gabor Filter and Centre Symmetric-Local Binary Pattern Based Technique for Forgery Detection in Images," *Multimedia Tools and Applications*, vol. 83, no. 17, pp. 50157–50195, 2024, doi: 10.1007/s11042-023-17485-1.

[16] C. Tan, Y. Zhao, S. Wei, G. Gu, P. Liu, and Y. Wei, "Frequency-Aware Deepfake Detection: Improving Generalizability Through Frequency Space Domain Learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 5, pp. 5052–5060, 2024, doi: 10.1609/aaai.v38i5.28310.

[17] S. Fang, Z. Zhang, and B. Song, "Deepfake Detection Model Combining Texture Differences and Frequency Domain Information," *ACM Transactions on Privacy and Security*, no. 21, pp. 1-16, 2024, doi: 10.1145/3697336.

[18] X. Jin, N. Wu, Q. Jiang, Y. Kou, H. Duan, P. Wang, and S. Yao, "A Dual Descriptor Combined with Frequency Domain Reconstruction Learning for Face Forgery Detection in Deepfake Videos," *Forensic Science International: Digital Investigation*, vol. 49, 2024, doi: 10.1016/j.fsidi.2024.301747.

[19] R. Tolosana, S. Romero-Tapiador, J. Fierrez, and R. Vera-Rodriguez, "XceptionNet: Advanced Deepfake Detection Through Multi-Attention Mechanisms," *IEEE Transactions on Information Forensics and Security*, vol. 40, pp. 1-12, 2025, doi: 10.1109/TIFS.2025.3989651.

[20] R. N. Bharath Reddy, T. V. Naga Siva, B. S. Ram, K. N. Ramya Sree and B. Suvarna, "Enhanced Deep Fake Image Detection via Feature Fusion of EfficientNet, Xception, and ResNet Models," *2025 6th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)*, pp. 1547-1552, 2025, doi: 10.1109/ICMCSI64620.2025.10883356.

[21] Z. Xia, T. Qiao, and M. Xu, "Deepfake Video Detection Based on MesoNet with Preprocessing Module," *Symmetry*, vol. 14, no. 5, pp. 939-952, 2024, doi: 10.3390/sym14050939.

[22] S. S. Khalil, S. M. Youssef, and S. N. Saleh, "Capsule-Forensics: An Integrated Approach for Deepfake Detection Using Dynamic Routing Between Capsules," *Future Internet*, vol. 13, no. 4, pp. 93-108, 2024.

[23] H. Ilyas, A. Javed, and K. M. Malik, "Two-Stream Neural Network for Deepfake Detection: Combining Spatial and Temporal Features," in *2024 IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 1-8, 2024.

[24] S. Kingra, N. Aggarwal, and N. Kaur, "SFormer: An End-to-End Spatio-Temporal Transformer Architecture for Deepfake Detection," *Forensic Science International: Digital Investigation*, vol. 44, 2024, doi: 10.1016/j.fsidi.2024.301584.

[25] M. Zou, B. Yu, Y. Zhan, S. Lyu and K. Ma, "Semantics-Oriented Multitask Learning for DeepFake Detection: A Joint Embedding Approach," in *IEEE Transactions on Circuits and Systems for Video Technology*, doi: 10.1109/TCSVT.2025.3572508.

[26] A. M, K. S. Charan, S. BN and S. Kanmani R, "Deep Fake Detection using Transfer Learning: A Comparative study of Multiple Neural Networks," *2024 International Conference on Signal Processing, Computation, Electronics, Power and Telecommunication (IConSCEPT)*, pp. 1-6, 2024, doi: 10.1109/IConSCEPT61884.2024.10627869.

[27] G. Naskar, S. Mohiuddin, S. Malakar, E. Cuevas, and R. Sarkar, "Deepfake Detection Using Deep Feature Stacking and Meta-Learning," *Heliyon*, vol. 10, no. 4, 2024.

[28] N. M. Alnaim *et al*., "DFFMD: A Deepfake Face Mask Dataset for Infectious Disease Era With Deepfake Detection Algorithms," *IEEE Access*, vol. 11, pp. 16711–16722, 2023, doi: 10.1109/ACCESS.2023.3246661.

[29] A. Ciamarra, R. Caldelli, F. Becattini, L. Seidenari, and A. Del Bimbo, "Deepfake Detection by Exploiting Surface Anomalies: The SurFake Approach," *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1024–1033, 2024.

[30] G. H. Ishrak, Z. Mahmud, M. Farabe, T. K. Tinni, T. Reza, and M. Z. Parvez, "Explainable Deepfake Video Detection Using Convolutional Neural Network and CapsuleNet," *arXiv*, 2024, doi: 10.48550/arXiv.2404.12841.

[31] A. S. A. Al-Qazzaz, P. Salehpour, and H. S. Aghdasi, "Robust DeepFake Face Detection Leveraging Xception Model and Novel Snake Optimization Technique," *Journal of Robotic and Control (JRC)*, vol. 5, no. 5, pp. 1444–1456, 2024, doi: 10.18196/jrc.v5i5.22473.

[32] T. Qiao, S. Xie, Y. Chen, F. Retraint and X. Luo, "Fully Unsupervised Deepfake Video Detection Via Enhanced Contrastive Learning," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 7, pp. 4654-4668, 2024, doi: 10.1109/TPAMI.2024.3356814.

[33] J. Zheng, Y. Zhou, X. Hu, and Z. Tang, "Deepfake detection with combined unsupervised-supervised contrastive learning," in *Proc. 2024 IEEE Int. Conf. Image Process. (ICIP)*, pp. 787–793, 2024, doi: 10.1109/ICIP51287.2024.10647603.

[34] C.-Y. Hong, Y.-C. Hsu, and T.-L. Liu, "Contrastive learning for DeepFake classification and localization via multi-label ranking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17627–17637, 2024, doi: 10.1109/CVPR52733.2024.01669.

[35] D. Dagar and D. K. Vishwakarma, "A Hybrid Xception-LSTM Model with Channel and Spatial Attention Mechanism for Deepfake Video Detection," *2023 3rd International Conference on Mobile Networks and Wireless Communications (ICMNWC)*, pp. 1-5, 2023, doi: 10.1109/ICMNWC60182.2023.10435983.

[36] T. Kanwal, R. Mahum, A. AlSalman, M. Sharaf, and H. Hassan, "Fake speech detection using VGGish with attention block," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2024, no. 35, 2024, doi: 10.1186/s13636-024-00348-4.

[37] A. Khormali and J.-S. Yuan, "ADD: Attention-based DeepFake detection approach," *Big Data Cogn. Comput.*, vol. 5, no. 49, pp. 1–15, 2021, doi: 10.3390/bdcc5040049.

[38] Y. Liu, Y. Chen, W. Dai, M. Gou, C. -T. Huang and H. Xiong, "Source-Free Domain Adaptation With Domain Generalized Pretraining for Face Anti-Spoofing," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 8, pp. 5430-5448, 2024, doi: 10.1109/TPAMI.2024.3370721.

[39] X. Zhang, J. Yi, C. Wang, C. Y. Zhang, S. Zeng, and J. Tao, "What to remember: Self-adaptive continual learning for audio deepfake detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 17, pp. 19569–19577, 2024, doi: 10.1609/aaai.v38i17.29929.

[40] H. Liu, Z. Tan, C. Tan, Y. Wei, J. Wang, and Y. Zhao, "Forgery-aware adaptive transformer for generalizable synthetic image detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10770–10780, 2024.

[41] J. Zakkam, "CoDeiT: Contrastive Data-Efficient Transformers for Deepfake Detection," in *Lecture Notes in Computer Science*, pp. 62–77, 2024, doi: 10.1007/978-3-031-78125-4_5.

[42] C. de Weever, S. Wilczek, C. de Laat, and Z. Geradts, *Deepfake detection through PRNU and logistic regression analyses*, Technical report, University of Amsterdam, 2020.

[43] M. S. Rana, M. N. Nobi, B. Murali and A. H. Sung, "Deepfake Detection: A Systematic Literature Review," in *IEEE Access*, vol. 10, pp. 25494-25513, 2022, doi: 10.1109/ACCESS.2022.3154404.

[44] S. Dreiseitl and L. Ohno-Machado, "Logistic regression and artificial neural network classification models: a methodology review," in *Journal of Biomedical Informatics*, vol. 35, no. 5-6, pp. 352–359, 2002, doi: 10.1016/S1532-0464(03)00034-0.

[45] S. Solaiyappan and Y. Wen, "Machine learning based medical image deepfake detection: A comparative study," in *Machine Learning with Applications*, vol. 8, pp. 100298, 2022, doi: 10.1016/j.mlwa.2022.100298.

[46] M. S. M. Altaei and others, "Detection of Deep Fake in Face Images

Using Deep Learning," in *Wasit Journal of Computer and Mathematics Science*, vol. 1, no. 4, pp. 60–71, 2022, doi: 10.31185/wjcms.v1.i4.7.

[47] A. H. Setyaningrum, A. E. Saputro, and others, "Deepfake Video Classification Using Random Forest and Stochastic Gradient Descent with Triplet Loss Approach Algorithm," in *Proc. 2024 12th International Conference on Cyber and IT Service Management (CITSM)*, pp. 1–6, 2024, doi: 10.1109/CITSM58677.2024.10471234.

[48] N. Chakravarty and M. Dua, "A lightweight feature extraction technique for deepfake audio detection," in *Multimedia Tools and Applications*, vol. 83, pp. 67443–67467, 2024, doi: 10.1007/s11042-024-18217-9.

[49] M. T. Islam, I. H. Lee, A. I. Alzahrani, and K. Muhammad, "MEXFIC: A meta ensemble eXplainable approach for AI-synthesized fake image classification," *Alexandria Engineering Journal*, vol. 116, pp. 351–363, 2025, doi: 10.1016/j.aej.2024.12.031.

[50] K.-H. Moon, S.-Y. Ok, and S.-H. Lee, "SupCon-MPL-DP: Supervised Contrastive Learning with Meta Pseudo Labels for Deepfake Image Detection," *Applied Sciences*, vol. 14, no. 8, pp. 3249, 2024, doi: 10.3390/app14083249.

[51] J. Laakkonen, *Domain-Augmented Meta-Learning for Generalizable Deepfake Detection*, University of Eastren Finland, 2024.

[52] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "Faceforensics++: Learning to detect manipulated facial images," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1–11, 2019, doi: 10.48550/arXiv.1901.08971.

[53] L. Li, J. Bao, T. Zhang, H. Yang, D. Chen, F. Wen, and B. Guo, "Face X-ray for more general face forgery detection," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5001–5010, 2020.

[54] Y. Li, M.-C. Chang, and S. Lyu, "In ictu oculi: Exposing AI-generated fake face videos by detecting eye blinking," *arXiv*, 2018, doi: 10.48550/arXiv.1806.02877.

[55] G. Ciocca and R. Schettini, "An innovative algorithm for key frame extraction in video summarization," *Journal of Real-Time Image Processing*, vol. 1, pp. 69–88, 2006, doi: 10.1007/s11554-006-0001-1.

[56] X. Liu, Y. Yu, X. Li and Y. Zhao, "MCL: Multimodal Contrastive Learning for Deepfake Detection," in *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 34, no. 4, pp. 2803-2813, 2024, doi: 10.1109/TCSVT.2023.3312738.

[57] T. Qiao, S. Xie, Y. Chen, F. Retraint and X. Luo, "Fully Unsupervised Deepfake Video Detection Via Enhanced Contrastive Learning," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 7, pp. 4654-4668, 2024, doi: 10.1109/TPAMI.2024.3356814.

[58] B. Wang, X. Yue, Y. Liu, K. Hao, Z. Li, and X. Zhao, "A Dynamic Trust Model for Underwater Sensor Networks Fusing Deep Reinforcement Learning and Random Forest Algorithm," *Applied Sciences*, vol. 14, no. 8, pp. 3374, 2024, doi: 10.3390/app14083374.

[59] T. J. Reddy, M. S. Ganesh, M. H. Kumar Reddy, C. Bhandhavya and R. Jansi, "Deep Learning-Powered Face Detection and Recognition for Challenging Environments," *2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT)*, pp. 1453-1459, 2024, doi: 10.1109/IDCIoT59759.2024.10467753.

[60] Z. Yan, Y. Zhang, X. Yuan, S. Lyu, and B. Wu, "DeepfakeBench: A Comprehensive Benchmark of Deepfake Detection," *arXiv*, 2023, doi: 10.48550/arXiv.2307.01426.

[61] S. Fung, X. Lu, C. Zhang, and C.-T. Li, "Deepfake detection via unsupervised contrastive learning," in *2021 international joint conference on neural networks (IJCNN)*, pp. 1–8, 2021, doi: 10.48550/arXiv.2104.11507.

[62] B. N. Subudhi and others, "Adaptive Meta-Learning for Robust Deepfake Detection: A Multi-Agent Framework to Data Drift and Model Generalization," in *arXiv*, 2024, doi: 10.48550/arXiv.2411.08148.

[63] Y.-K. Lin and T.-Y. Yen, "A Meta-Learning Approach for Few-Shot Face Forgery Segmentation and Classification," in *Sensors*, vol. 23, no. 7, pp. 3647, 2023, doi: 10.3390/s23073647.

[64] J. Zakkam, "CoDeiT: Contrastive Data-Efficient Transformers for Deepfake Detection," in *Lecture Notes in Computer Science*, pp. 62–77, 2024, doi: 10.1007/978-3-031-78125-4_5.

[65] S. Fung, X. Lu, C. Zhang, and C.-T. Li, "Deepfake detection via unsupervised contrastive learning," in *2021 international joint conference on neural networks (IJCNN)*, pp. 1–8, 2021, doi: 10.48550/arXiv.2104.11507.

[66] L. Baraldi, "Contrasting Deepfakes Diffusion via Contrastive Learning," *Computer Vision – ECCV 2024*, pp. 199–216, 2024, doi: 10.1007/978-3-031-73036-8_12.

[67] G. Naskar, Sk. Mohiuddin, S. Malakar, E. Cuevas, and R. Sarkar, "Deepfake detection using deep feature stacking and meta-learning," *Heliyon*, vol. 10, no. 4, 2024, doi: 10.1016/j.heliyon.2024.e25933.

[68] M. M. Ghazi and H. K. Ekenel, "Performance Analysis on Deep Fake Detection," *IBIMA Publishing*, vol. 2024, 2024, doi: 10.5171/2024.457767.

[69] D. Wodajo, S. Atnafu, and Z. Akhtar, "Deepfake video detection using generative convolutional vision transformer," *arXiv*, 2023, doi: 10.48550/arXiv.2307.07036.

[70] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.