Multimedia Tools and Applications
https://doi.org/10.1007/s11042-024-20268-x

Regularized forensic efficient net: a game theory based generalized approach for video deepfakes detection

Qurat Ul Ain[1] · Ali Javed[2] · Khalid Mahmood Malik[3] · Aun Irtaza[1]

Received: 19 February 2024 / Revised: 27 August 2024 / Accepted: 11 September 2024

Abstract
Due to the advancements in cutting-edge generative AI algorithms, generating hyper realistic deepfake videos has become easier for the public. This hyperrealism consequently fails contemporary methods to reliably discriminate between original and fake videos. Therefore, to counter any threat caused by these next-generation artificially generated videos, dependable approaches are required to address this classification challenge. To achieve this objective this paper presents an interdisciplinary approach that integrates game theory with deep learning to bring a novel solution to the problem of deepfake detection and protect the detectors against anti-forensics attack. To the best of our knowledge, there does not exist any other work dedicated to video deepfake detection using the integrated approach of game theory and deep learning. The game is designed for two players to distinguish between pristine and deepfake videos. The game utilizes different strategies for the data manipulator as a player P_1 and the deepfake detector as P_2. Strategies used for P_1 involve the formation of the subsets like open and close-set, combined subsets, imbalanced dataset, and post-processing attacks to create challenging strategies for P_2. To counter the strategies of P_1, we propose a novel Regularized Forensic Efficient Net (RFE Net) that employs regularization techniques, such as batch normalization, dropout, augmentation, and early stopping. Based on the P_1 move, the detector chooses the regularization techniques by considering factors such as generalizability and efficiency. Regularization-based strategies improve the performance of our model when compared to contemporary methods. Computation of the Nash equilibrium with the proposed zero-sum game helps to effectively detect deepfakes and leads the game to maximum payoff. Performance of the proposed game theory-based RFE Net was measured on standard and diverse datasets of FaceForensic++, DFDC preview, CelebDF, DFFD, and the World leader's dataset, including cross-set and cross-corpus evaluation. Additionally, we also performed the post-processing attacks evaluation and explainability analysis. Experimental results demonstrate that the proposed RFE Net outperforms state-of-the-art methods for deepfakes detection in the defined premises.

Keywords Deepfakes · Game theory · Nash equilibrium · Post-processing attacks · Regularized forensics efficient net · Zero-sum game

Extended author information available on the last page of the article

1 Introduction

Recent years have witnessed the exponential growth of multimedia content such as images, videos, and audio, on the internet due to the easy availability of economical gadgets like smartphones and digital cameras. According to a survey [1], up to 4 to 5 billion posts are shared on social media platforms daily, and most of these are images and videos. With the introduction of sophisticated AI algorithms that can easily manipulate vast quantities of multimedia content, spurred by considerable progress in the field of generative AI, fake images and videos can be created in minutes. AI-based algorithms like generative adversarial networks (GANs) [2], diffusion models [2], and automatic image and video editing tools such as Photoshop have made it convenient to modify and generate fake social media content in a noticeably short time. The massive increase in fake content leads to the spread of disinformation and misinformation online via social media platforms.

In the last few years, the term "Deepfake" has become more familiar because of social and electronic media platforms. Deepfakes are videos or audio generated from deep learning algorithms to create synthesized content, usually with the intent to deceive. Deepfakes can be used to spread disinformation and misinformation around the world and pose a serious new threat in the form of falsified "evidence". The use of robust and easy-to-use manipulation tools, like REFACE [3], DeepFaceLab [4], and others, makes the authentication and integrity verification of electronic media even more difficult. Identity and expression swapping, and lip-syncing are a few of the diverse types of deepfakes. Identity swapping (Faceswap, Face shifter, and DeepFakes) [5] exchanges the faces of the source and target person using a deep learning-based or graphics-based manipulation like graphic rendering [6]. Deepfakes with face swapping are frequently used to harm someone's reputation. In lip-sync-based [7] deepfakes, the lip movement of the targeted person is adjusted to match a specific audio recording. Attackers use lip-syncing to give the impression that their target is speaking the falsified audio that accompanies such clips. Expression swapping [8] includes the face2face [9] and neural texture methods. In this form of deepfake, an impersonator may manipulate the source person's expression, but also, in more extreme examples, complete body movements during image or video modification. Society is at risk because malicious actors use deepfakes to create false profiles to disseminate disinformation on social media. In addition to bolstering belief in deepfake-created content, the development of compelling deepfake apps has frustrated world leaders and celebrities who are often the targets of such attacks. Deepfakes are becoming increasingly difficult to spot on social media because modern-day sophisticated techniques are good enough to fool an uninformed public.

Different approaches for video deepfake detection have been used, including handcrafted features-based with back-end ML or DL classifiers, end-to-end DL, and DL features extractor-based with conventional ML classifiers. ELA [10], SURF [11], MDHFD [12] techniques are used in pre-processing. These descriptors extract useful features to train a traditional ML classifier like an SVM, but they are computationally expensive. End-to-end learning uses DL to extract and train features between input and output layers. Convolutional neural networks (CNN) train all parts simultaneously. Pre-trained models like ResNet, Inception Net, etc., extract deep features and train conventional ML algorithms during DL feature extraction. DL-based deepfake detectors like fused models [13] and vision transformers [14] are computationally complex. Few research [13–15] employs various regularisation techniques for deepfake detection; however, we present an integrated approach of game theory and deep learning. Although various regularisation techniques

are effective in deepfake detection, a game theory approach that incorporates regularization has the potential to provide more comprehensive and resilient detection capabilities. All the above-mentioned deepfake detection techniques need further improvements in effectiveness, efficiency, robustness to different postprocessing attacks, and generalizability.

In recent years, game theory has received considerable attention in the field of computer science [16], particularly in its application to artificial intelligence, neural networks [17], and deep learning [18]. Game theory is a method of analyzing strategic interactions between players with self-interest. Because of this, it has significant applications in economics and a variety of other fields, as well as computer science. All these disciplines share an interest in strategic decision-making and determining the optimal structure for these interactions. Game theory is the study of strategies to explain the interaction between players to find an optimal solution to achieve a specific outcome. Emile Borel was perhaps the first mathematician to organize a system for playing games. The concept of probabilistic game theory in strategic games of chance was then properly introduced by Jhon Von Neumann and Oskar Morgenstern in 1928 [19]. Together they made advances in the field of research based on game theory. Based on Neumann's research, the min-max theorem for a two-player, zero-sum game with pure strategies was introduced. Min-max specified a required outcome and a payoff in the form of a win or a loss. In a zero-sum game, only these two outcomes are defined. Game theory investigates how intelligent players should act in a given scenario to maximize their payoffs. One player receives an input, which is the current state of the other player. The strategy then chooses an action that changes the present state, and the value of the state transition is communicated to the player, called a payoff. The players learn the policy through continuous trial and error.

The motivation of this research is to devise an interdisciplinary approach involving game theory with deep learning. The existing deepfake detectors mostly used hand-engineered or deep learning techniques, which are not robust and lack generalizability in several conditions like cross-set scenarios. The contemporary methods are sensitive to post-processing attacks like noise and blurriness. In addition, the existing methods used DL techniques as a black box model, which lacks the explainability factor. To tackle all these problems, the proposed research aims to provide a generalizable and explainable deepfake video detection approach based on a game theory idea in which the data manipulator and detector are considered as two players in a zero-sum game. The game is designed for two players to distinguish between pristine and deepfake videos. The game utilizes different strategies for the data manipulator as a player P_1 and the deepfake detector as P_2. Strategies used for P_1 involve the formation of subsets like open and close-set, combined subsets, imbalanced datasets, and post-processing attacks to create challenging strategies for P_2. To counter the strategies of player P_1, we propose player P_2, a novel Regularized Forensic Efficient Net (RFE Net) a game theory-based deepfake detector that employs various regularization techniques, such as batch normalization, dropout, augmentation, and early stopping, with a deep neural network. Players follow mixed random strategies to achieve better outcomes and maximize the Nash equilibrium (NE), a state where players reach their desired goal. Strategy plays a pivotal role in achieving the optimal outcome. To overcome limitations in the existing approaches, the proposed model can detect several types of video forgeries, such as identity swapping (face-swap, deepfakes), expression swapping (face2face, neural textures), lip-syncing, and the comedic impersonator. To maximize the payoff of the Nash state, the proposed RFE Net architecture uses regularization-based strategies. To our knowledge, this unique approach utilizes the principles of game theory to analyze and identify deepfake videos accurately. The proposed research improves the generalizability of the model in contrast to contemporary CNN-based deep fake detection algorithms.

The following are the main contributions of this research:

- We propose a novel Regularized Forensic Efficient Net model, a combination of supervised learning and game theory for video deepfake detection.
- Proposed RFE Net can accurately identify several types of deepfakes, including expression, identity swap, and lip-syncing. Using regularization techniques in the proposed model improves the robustness and generalizability of the model and helps combat post-processing attacks. Moreover, we employ the ELU activation function to prevent neurons from dying, regularity for better gradient flow and accelerated learning.
- We employ a zero-sum game for two players, with different strategies for both players (P_1 and P_2) to check the generalizability of the proposed RFE Net (P_2). The game achieves a payoff by maximizing the Nash equilibrium based on P_2 results.
- Using mixed strategies leads our model to classify between a deepfake and a pristine video and assign rewards to players on NE using two-player game scenarios.
- We performed rigorous experimentation on benchmark datasets and compared the proposed method against contemporary methods, including cross-set and cross-corpus experiments, imbalanced datasets, and post-processing attacks evaluation to show the generalizability and robustness of the proposed method.

The paper is organized into the following sections: Section 2 discusses contemporary methods for deep fake detection and game theory. Section 3 presents the details of the game theory-based RFE Net. The details of datasets and experimentation results are provided in Section 4, and Section 5 presents the conclusion and future directions.

2 Literature review

This section provides a comprehensive discussion of contemporary methods to identify video deepfakes and game theory approaches based on classification and detection. Video deepfake manipulation techniques fall into two major types: (i) graphic-based methods, and (ii) learning-based methods [20]. Faceswap and face2face are graphic-based manipulation methods; learning-based methods include the deepfake, neural textures, lip-sync, and face shifter. According to deepfake detection: a systematic literature review [21], deepfake detection techniques are grouped into the following types: hand-crafted features-based with back-end ML or DL classifiers, end-to-end DL-based, and DL feature extractor-based with the conventional ML classifiers. We divided the literature into sub-sections for a detailed review of deepfake detection and a discussion of the techniques, and the limitations associated with the existing methods.

2.1 Contemporary methods for deepfake detection

This section reviews the deepfake manipulation types discussed above along with their detection strategies in detail, but we further group detection techniques into two subcategories: deepfake detection based on hand-engineered features [10–12, 19–34] and deepfake detection based on ML and DL [28–45].

2.1.1 Deepfake detection based on hand-engineered features and traditional machine learning-based classifiers

In [10–12, 19–34], hand-engineered deepfake detection techniques are discussed. Zhang et al. [11] used SURF for feature extraction and trained these features using an SVM for face swap deepfake detection. This method [11], however, is only robust to static frames as it is unable to detect frames with sudden changes. Yang et al. [22] proposed a method for detecting facial landmark features of AI-generated faces that used these extracted features to train the SVM. This technique is unable to achieve better deepfakes detection performance on blurred and fuzzy facial frames. Jack et al. [23] employ multimedia stream descriptors for feature extraction and used them to train an SVM with the extracted features for face swap detection but this method is ineffective for operations related to video re-encoding. Ciftci et al. [24] proposed a method using biological signals for face swap detection by computing forensic changes in facial expressions, which were then used with a CNN and SVM. Performance of this technique degrades when the image dimensions are reduced. Jung et al. [25] proposed a method for detecting deepfake inconsistencies from video samples based on time, repetition, and eye-blinking, using Fast-HyperFace and EAR, however, this technique fails to detect frames with frequent eye blinking. McCloskey et al. [26] presented image color key-point features for the detection of GAN-generated images and used them to train an SVM classifier. This approach discriminates between real and fake images accurately based on color, but the performance of this method degrades on blurred images. In our prior work [10], we used both handcrafted and DL techniques. Error Level Analysis was used for feature extraction and different DL models were trained on these features. The models include VGG-16, VGG-19, Inception-V3, and Resnet 50 for synthetic face detection but this method [10] needs improvement in terms of accuracy.

Guarnera et al. [27] proposed an Exception Maximization method for feature extraction and use these features to train a KNN and an SVM for deepfake detection but this method shows degraded performance when applied to compressed images. Nataraj et al. [28] proposed a feature extraction method based on pixel co-occurrence matrices. A CNN was trained on extracted features for the determination of real and deepfake samples, but this method fails to detect noisy samples. Zhang et al. [29] proposed a GAN-generated deepfake detection method using 2D DFT key-point extracted features from the frequency domain. This method is unable to perform well on some GAN-generated images, like GuaGAN-generated realistic images. Amerini et al. [30] proposed a face-2face (F2F) deepfake detection approach based on optical flow features and trained with a CNN, but the accuracy of this method needs improvement. Agarwal et al. [31] used the OpenFace2 toolbox [32] for the 2D/3D identification of facial and head landmarks and trained an SVM with those features. This approach, however, is unable to detect the face of an individual looking off-camera. Korshunov et al. [33] proposed a technique for lip-syncing deepfake detection with different classifiers, an SVM, LSTM, multilayer perceptron (MLP), and a Gaussian mixed model (GMM), trained using Mel-scale Frequency Cepstral Coefficients (MFCC). Performance of this method varies on different datasets and is poor on datasets with fewer training samples. Boutellaa et al. [34] used DNN-based phonetic features for lip-sync-based speaker identification but the performance of this method degrades when detecting on a GAN-generated dataset. In our previous research [12], we developed MDHFD, which combined LHeXDP and LANMP to extract directional and magnitude details from adjacent pixels used to classify deepfake

videos as original or face-swapped. This method [12] performed better than existing techniques in detecting face-swap, but it is computationally more complex.

2.1.2 Deepfake detection based on deep features

In works [13–15, 35–61], deepfake detection techniques based on deep learning are discussed. Li et al. [35] used DLib [36] facial landmark features and trained different pretrained deep learning models like VGG-16, ResNet, etc., with extracted features, for faceswap deepfake detection. This technique attains good results, but performance degrades on compressed videos. Guera et al. [37] proposed a CNN model for feature extraction and used the extracted features to train an RNN but this technique fails to detect face swap frames from longer video samples. Nirkin et al. [38] developed a technique to identify identity manipulation using face recognizer confidence scores. The model was trained on cropped images to get face recognition scores. A deep XceptionNet detected the facial identity modifications after scoring. The XceptionNet is incapable of identifying deepfake artifacts. Face recognition makes the model resource-intensive and inapplicable to unknown images. In [39], videos are considered to be time-series data to capture the interconnections of individual frames. The depth-separable convolution layers of the Xception network are utilized to train the model. Liy et al. [40] proposed a CNN/RNN-based method to capture spatiotemporal features to detect deepfakes through eye-blinking in AI, or machine-generated, faces. This technique fails to detect deepfakes in videos with persistent eye blinking or closed eyes. Montserrat et al. [41] proposed a method for video face-swap detection that uses the Automatic Face Weighting (AFW) technique to delete non-face frames. Extracted AFW features were then used to train a CNN and an RNN for deepfake detection, but the accuracy of this method needs to be improved. Lima et al. [42] proposed a method for face swap detection through the computation of spatial features extracted from a VGG-11 and temporal features using an LSTM. Different models, CNN, R3D, ResNet, and I3D, were trained on the extracted features, but this method [42] is computationally more complex. Guarnera et al. [27] proposed a technique for identifying deepfakes using a deep learning-based Exception-minimization model for feature extraction. Then, this method used the extracted features to train a Naïve Bayes classifier. The method is only robust on static images, and the performance of this method degrades on compressed frames. Agarwal et al. [43] proposed a method for detecting face-swap-based manipulations, detecting facial traits using VGG-16 and behavioral biometrics through Facial Attributes-Net [44]. This method is imprecise when detecting unseen samples. Jian et al. [44] proposed a CNN-based, key points feature extractor with an SVM classifier for deepfake detection. This technique fails to accurately identify deepfakes on noisy and blurred samples. Rathgeb et al. [45] proposed a Photo-Response Non-Uniformity (PRNU) technique to extract spatial features for the detection of facial attribute manipulation samples but this method experiences reduced performance in unseen scenarios. Tariq et al. [46] used different deep learning models, such as VGG-19, VGG-16, ResNet, and XceptionNet, for video facial attribute manipulation detection but this model failed to detect real-world samples. Heidari et al. [47] proposed a new federated learning system based on blockchain that protects the privacy of data sources and uses techniques like SegCaps, CNN, capsule network training, transfer learning, and preprocessing. Experiments show a 6.6% accuracy increase and 5.1% AUC improvement, requiring further research for practical use.

Marra et al. [48] proposed a deep learning model called XceptionNet for the identification of facial manipulation. Although this method has shown good performance it

has a higher computational cost, and the system fails to detect forgeries in cases where source manipulation information is missing. Afchar et al. [49] proposed CNN-based models named MesoNet, Meso-4, and the fusion of Inception Net and Meso-4. The Meso-Inception-4 model was trained for detection on the face2face and deepfake subsets of FaceForensic++(FF++), but the performance of this technique degrades when used on the low-quality video. Sabir et al. [50] used an RNN to deal with temporal distortion for synthetic face detection but this technique is limited to the detection of static images only. Nguyen et al. [51] introduced a CNN network for recognizing and localizing changed video content that is multitasking and based on machine learning. Using an autoencoder to classify forgery and a y-shaped decoder to forward the extracted information for classification and reconstruction. Despite being resistant to deepfake detection, this model's accuracy decreases when presented with an unanticipated scenario. Seraj et al. [52] introduce a deep domain adaptation framework for detecting deepfakes using labelled data from faking techniques. It optimizes loss and trains the network using a novel loss function and stochastic gradient descent. However, it relies on annotated data and struggles to identify emerging techniques. Matern et al. [53] proposed a CNN strategy for detecting deepfake based on visual anomalies. Due to the large size of the feature space, this method experiences a high computational cost. Haliassos et al. [54] proposed a spatiotemporal network-based model for detecting lip-sync deepfakes. For lip-reading, 3D-CNN and ResNet18 models were used, and a multiscale temporal convolutional network was employed to extract deep features. The model is then fine-tuned and effective in the presence of post-processing techniques such as blurring, noise, compression, etc. It performs substantially worse, however, when mouth movement is restricted, such as during pauses in speech. Das et al. [15] proposed dynamic face augmentation, which may not capture the full diversity of real-world scenarios, leading to potential issues in generalization. Additionally, the computational demands of the deep learning techniques might limit practical applicability, and the model could struggle with detecting advanced or novel deepfake methods not covered in the training data. Generalization of deep Q-network DQN is proposed in [13], where regularization prevents overfitting to training data. Balancing these aspects is challenging, as over or under-applying regularization affects the model's learning ability. Achieving this balance often requires extensive hyperparameter tuning and domain expertise. The inherent limitations in DQN's adaptability to vastly different environments from training data remain a significant challenge. Cheng et al. [14] presented a concept of primary region regularization, which involves identifying and analyzing specific regions within a video frame that are most likely to contain manipulated content or artifacts. However, this technique is limited to other forms of synthetic media manipulation, such as audio deepfakes or manipulated images.

Demir et al. [55] developed a deepfake source detector that outperforms previous models by 4.08% in identifying fake videos and their source generators but faces limitations in identifying advanced deepfakes. Bonettini et al. [56] investigated how utilizing several different CNN models impacted the performance of an ensemble classifier. Ilyas et al. [57] proposed a novel efficient-capsule network (E-Cap Net) for forgery detection. Capsule network architecture improves image and video deepfake detection, however, this technique is computationally more complex. Khalid et al. [58] proposed a fused truncated DenseNet121 model to detect deepfakes through transfer learning, truncation, and feature fusion. The model accurately detects deepfakes in diverse datasets. Ilyas et al. [59] proposed a combined Swish and ReLU activation functions to improve the representation capabilities of the Efficient-Net architecture for deepfakes detection. The model achieved good performance on deepfake detection, however, it lacks a comprehensive discussion

on the generalization of the model to unseen deepfake variations. These two methods [58, 59] lack generalizability in cross-corpus experiments. In our previous research [60], we proposed DFGNN, an interpretable and generalized GNN for deepfake detection. It uses facial landmarks to create a graph, improving interpretability and generalizability. However, advanced deepfakes may affect DFGNN's performance. More research is needed to evaluate its effectiveness in detecting complex deepfakes. Structurally regular advanced deepfakes may affect DFGNN's performance. However, there is a need for a more robust and generalizable method to detect complex synthetic content. Raza et al. [61] proposed a method that combined spatiotemporal transformer embeddings with a CNN architecture to detect deepfake videos with high accuracy. The method included long-range dependencies and contextual information, making it resistant to sophisticated deepfake manipulations. However, a notable limitation was the computational complexity of the spatiotemporal transformer embeddings, which could have hindered real-time deployment on resource-constrained platforms.

2.2 Existing methods based on game theory

This section provides an overview of state-of-the-art techniques [62–72] in the field of computer science based on game theory, specifically machine learning-based approaches. Several investigations [62, 63] have examined the game-theoretic concept underlying Deep Neural Network (DNN) representations. However, there is a lack of research in the field of detecting deepfakes using game theory. Dong et al. [64] provided a hypothesis through the evaluation of visual concepts on images using image matching from a new perspective based on Shapley values. Fernandes et al. [65] presented an adversarial attack on the deepfake detection system using a reinforcement learning-based texture patch attack method. To examine the complete potential of game theory in the context of deepfakes, however, additional research is required. In one of the recent works, Hazra et al. [18] provided a survey of the applications of game theory based on deep learning approaches. In this paper, the authors covered two major domains: GAN and reinforcement learning with aspects of game theory. Tembine proposed a game theory-based algorithm named Bregman [66] to show the relationship between GANs and game theory, in which the generator acts as one player and the discriminator as another, however, this method lacks explainability. Yasodharan et al. [67] introduced hypothesis testing of adversarial classification based on a non-zero-sum game theory approach. This work still needs further improvement when it comes to detection. Sanchez [68] proposed a strategic game, linked with binary classification, based on a zero-sum game using the min-max principle. Wu et al. [69] proposed a zero-sum game using the min-max phenomena for the binary classification of two players. The outcome of the method was calculated using GAP and mean. Couellan et al. [70] proposed a binary and multiclass classification technique introducing Nash and generalized Nash with an SVM. Georgiou et al. [71] proposed a cooperative and collation game using an ensemble of different classifiers, an SVM, KNN, and a Decision tree, for binary classification. Behpour et al. [72] presented a game-theoretic model for data augmentation using adversarial picture perturbation annotations. Experiments on the method [72] show that this model is capable of learning strong predictions under a wide range of augmentation sets for resistance to adversarial conditions, but the accuracy of this method needs to be improved.

There are many state-of-the-art methods dedicated to detecting deepfakes, and the concept of game theory has been employed in many fields of deep learning and CNN for

reinforcement and segmentation, among others. However, no attempt has been made, until now, to apply the game theory concept to deepfake detection. Game theory, which provides a decision-making framework, allows players to decide using multiple strategies. Combining the concepts of game theory and deepfake detection for video sample classification produces significantly more accurate results.

3 Proposed method

This section provides a comprehensive overview of game theory, and the Regularized Forensic Efficient Net model proposed for video deepfake detection. The subsequent sections present the details of our method.

3.1 Game theory

In this section, we illustrate the proposed method by designing a two-player game for detecting deepfakes. The game proposed for deepfake detection is an iterative zero-sum game. In a game, the data manipulator is considered one player and the deepfake detector the other. During the game, players can apply a variety of strategies to maximize the net NE. Therefore, the game can be modeled as an iterative or repeated game where each player learns and adapts their strategies over multiple interactions. A payoff matrix with "True Positive" (TP), "True Negative" (TN), "False Positive" (FP), and "False Negative" (FN) values represents the player's reward. Maximum TP and TN enable the game to attain NE.

The following terms are used regarding game theory:

- **Players**: $\boldsymbol{P}_1$: Data Manipulator
  $\boldsymbol{P}_2$: Deepfake Detector
- **Strategy**: Mixed Strategy (Detector Parameters).
- **Payoff matrix:** (max = TP, TN, min = FN, FP)
- **Nash equilibrium:** Best response.
- **Game Type**: Zero-sum game.

3.1.1 Players

In a game, players participate to achieve a reward. In our strategic game, the players are represented as P_i where $i = (1, \ldots, n)$. Each player P_i has a set of available actions A_i. In game theory, matrices are generally used to represent the game, and in the proposed game, the first player's actions are represented in rows, player two's actions are the columns of the matrix, and each cell in the matrix represents a possible resultant value. The utility of players for every outcome is recorded in the corresponding cell. In a generic game, the players can take a finite or infinite number of actions, but in the proposed strategic game, each player has a finite set of actions. The goal of each player is to maximize the payoff, and players $P_i$change their strategies during the game to reach their optimal outcomes. We define the proposed game as follows:

- We introduce a 2-player finite game, represented as $\left(P_i, Ai,\ u_i\right)$, where:

- P_i is a set of two players P_1 and P_2 indexed by $(i = 1, 2)$. Where P_1 is data manipulator and P_2 is deepfake detector.
- a_i is a finite set of actions available for P_i. Each vector $a_i = (a_1, \ldots, a_n) \in A$, where A is an action profile, such as manipulation of datasets and use of regularization etc. are the representative actions.
- $u = (u_1, \ldots, u_n)$, where u_i is payoff function created on bases of a_i.

3.1.2 Strategies

Strategy is defined as the series of actions specified by each player. Depending on the objectives of the players, there are two types of strategies, pure and mixed. A pure strategy mandates the player to select any action from the available set, while in a mixed strategy, a player can select a random action. In the proposed game, players use randomly available actions based on probability distribution.

A mixed strategy for a normal-form game is defined as follows: Let (P_i, A_i, u_i) be a game for any set Y, let $\prod (Y)$ be the set of all probability distributions. For P_i, the set of mixed strategies is defined as $Si = \prod (Ai)$. The Cartesian product of the mixed-strategy sets provides the collection of strategies, $(s_1 \times \cdots \times s_n)$. Any action a_i, played under mixed strategy s_i is represented as the probability of $s_i(a_i)$. The subset of actions that are assigned positive probability by the mixed strategy is called the support of s_i.

Player P1 Strategies The game starts with data manipulator P_1 turn making different strategies. It includes different types of deepfake: faceswap, face shifter, neural textures, face 2 faces, etc. In addition, imbalance, cross corpora, cross set, and post-processing attacked sets are also prepared by the data manipulator to create a tough game for the deepfake detector.

• Here is the P_1representative actions profile A ={combined test set, imbalanced set, cross corpora set, cross set, and post-processing attacks}.

Player P2 strategies In response to every strategy of the data manipulator, player P_2 deepfake detector employs its best strategy to detect deepfakes. In many cases P_2 overfits (without the use of regularization), making it computationally more complex as the training time of the P_2 increases, also class imbalance and attacks make the detection task more challenging. To minimize the overfitting issue and save training time for P_2, use regularization techniques like batch normalization, dropout, early stopping, and data augmentation. The selection of an activation function for better deepfake detection is also another strategy used by P_2 to achieve computational efficiency.

- Here is the P_2 actions profile A = {batch normalization, dropout, early stopping, and data augmentation}.

Overall, strategy plays a significant role in the game, and the strategy that leads the players to maximum payoff is considered best.

3.1.3 Nash Equilibrium

In the game theory, every game is based on a decision-making theorem called Nash equilibrium, which asserts that a player can attain the desired outcome by sticking to their initial strategy. Considering the decisions of other players, each P_i tries to achieve the optimal NE. In the proposed game-theoretic model, P_i utilizes a mixed strategy. When data manipulator (P_1) or deepfake detector (P_2) combines their tactics with uncertainty, the mixed equilibrium is calculated based on the predicted payoffs to the individual players. Our first insight is that if a P_2 knows how the P_1 will play, then P_2' s strategic dilemma is significantly simplified. Formally, mixed strategy is defined as$s_i = \left(s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n\right)$, and the strategy leads the game to NE is the best response s_i^*defined as.

• The best response for $P_i$to the strategy profile s_i is a mixed strategy $s_i^* \in s_i$ such that $u_i\left(s_i^*, s_{-i}\right) \geq u_i\left(s_i, s_{-i}\right)$ for all strategies $s_i \in S_i$.

When the player supports the best response for some actions, it must be agnostic to that action. However, any combined or individual action must constitute the best response. A player cannot predict what strategies the other players will employ, so the optimal response is a collection of possibilities. In case of post-processing attack (s_i) by P_2, the player P_1 used dropout and augmentation as best strategy (s_i^*), which helps to achieve Nash equilibrium. However, we can use the concept of optimum response to determine NE in non-corporative games. According to Nash's theorem:

- Every finite game has a Nash equilibrium.
- A NE is a strategy profile $s_i = \left(s_1, \ldots., s_n\right)$, , if s_i is the optimum response to s_{-i} for all agents i .

3.1.4 Pay-off matrix

The notion of mixed strategies based on fundamental decision theory is known as utility. We initiate by calculating the probability of each occurrence and then compute the average of the potential payoffs, weighted by the likelihood of each outcome. The formalization of the expected utility is as follows: (overloading notation, we use u_i for both utility and expected utility). Given a normal-form game $\left(P_i, Ai, u_i\right)$, the expected utility u_i for P_i of the mixed-strategy profile $s = \left(s_1, \ldots, s_n\right)$ is defined as:

$$u_i(\mathrm{s}) = \sum\nolimits_{a \in A} u_i(\mathrm{a}) \prod\nolimits_{j=1}^{n} s_j\left(a_j\right) \tag{1}$$

The payoff of a game is the incremental gain/benefit or loss/cost that a player earns because of executing its action against the strategy of the other player. Figure 1 shows the payoff matrix of our proposed method, where the red boundary around the upper left and lower right blocks represents the Nash state, while the remaining two represent the non-Nash state. The maximum outcome, TP, and TN lead our game towards the NE. The game planner creates the payoff function of the game. In the proposed game, the maximum TP, TN, and AUC represent the reward.

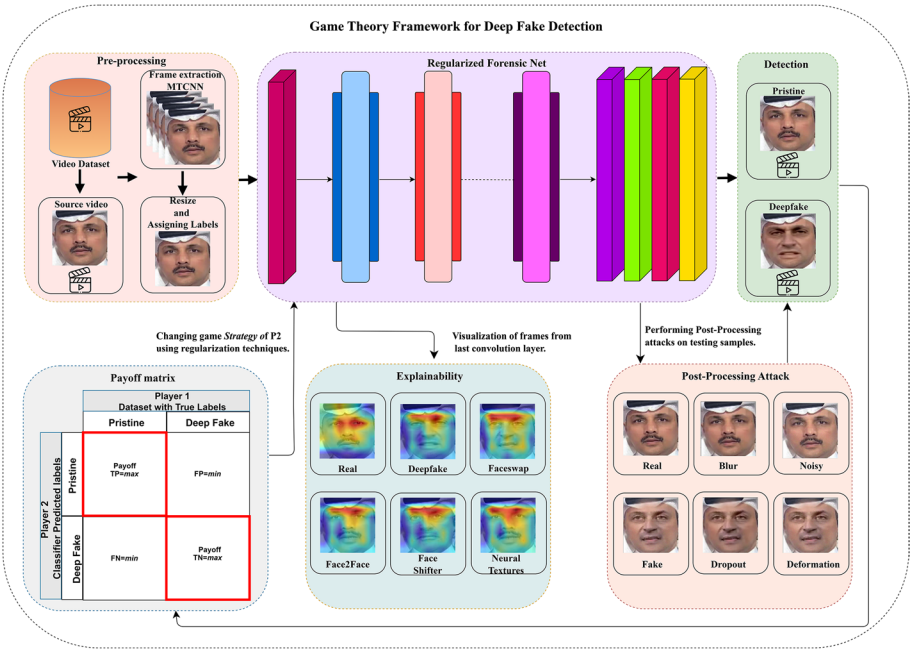

Fig. 1 Game Theory Framework for Deepfake Detection. The video dataset undergoes preprocessing where facial frames are extracted and trained using the proposed Regularized Forensic Net, accurately classifying them as real or deepfake. Game theory integration enables players to customize strategies, enhancing the system's generalizability. For model interpretability and explainability, heatmaps are produced. The model is also evaluated against postprocessing attacks to test its generalizability

3.1.5 Zero-sum game

Consider a detector as the player of interest in a zero-sum game to identify the best operating point using the min-max approach from game theory. In zero-sum games, one player's advantage must come at the expense of the other player in contrast to common-payoff games. In deepfake detection, the matrix for a zero-sum game correlates directly to the detector's confusion matrix. The game utilities are a, b, c, and d, where the detector has maximum TP and TN values leading to the utility value of $a = d = 1$, and minimum FN and FP correspond to $b = c = 0$. Maximizing the accuracy of the P_2 utility in a zero-sum game against P_1 is an identical scenario. The utility function of the proposed method is defined in Eq. (2).

$$Utility = \frac{a.TP + b.FP + c.FN + d.TN}{TP + FP + TN + FN} \tag{2}$$

3.2 Deepfake Detection Game

Using game theory for deepfake detection allows us to design the two-player game between P_1 and P_2. For this purpose, both players $P_1$and P_2 used mixed strategies (s_i) have different possibilities to achieve the payoff. P_1 in this game has the probability of pristine and deepfake samples, and P_2 has the probability of detecting these samples and resisting anti-forensic attacks. The mixed strategy of both players leads the game toward NE, and the outcome of a game having maximum TP and TN in the payoff matrix allows the game to achieve the best response payoff. Although the strategies of both players changed during the deepfake detection game, we consider P_2 as the player of interest who maximizes NE to minimize the non-Nash state.

The game begins with the P_1 turn passing different datasets, in response the pessimistic approach of P_2 leads our game to NE and achieves maximum payoff. Employing selected regularization techniques helps to minimize overfitting and save training time for P_2, it may also increase the accuracy and effectiveness of the deepfake detector in detecting more complex and sophisticated deepfakes. We further elaborate the strategies of both players in detail in the coming sections and the algorithm for the two-player game is mentioned in Algorithm1.

Assuming a game strategy where the attacker utilizes post-processing attacks ($s_1, s_2, \ldots, s_n$) on a deep learning model, but the defender applies regularization techniques ($s_1, s_2, \ldots, s_n$) to counteract these attacks. From all of the defender strategies s_i, dropout (s_1) enhances robustness by prohibiting reliance on specific neurons, which may reduce the efficacy of adversarial attacks that are specifically targeted. The model's capacity to identify and resist adversarial manipulations is improved by data augmentation (s_1), which broadens exposure to a variety of scenarios. The best regularization strategies s_i^* effectively decrease the benefit for the attacker and slightly increase the expense for the defender; equilibrium will probably be reached with both parties choosing their respective strategies. Dropout and data augmentation tactics improve a detector's robustness in Nash equilibrium, stabilizing the attacker and defender's best strategies. The detector's increased resistance may prevent attacks and strengthen the defensive approach as a stable, mutually optimal response by reducing the attacker's exploit attempt return.

3.3 Face detection

A Multi-Task Cascaded Convolutional Neural Network (MTCNN) [73] is used to extract frames from the video sample. Using facial landmarks like nose, eyes, and mouth, this model detects faces in input videos. Compared to other face detectors such as Viola-Jones and Haar-cascade [74], MTCNN isolates the face from the rest of the frame and extracts minute facial landmark details. MTCNN is capable of precisely detecting features and capturing coarse-to-fine detail, even under varying illumination and occlusion conditions. During preprocessing, we used MTCNN to detect individuals in each frame. After facial features have been extracted, images are resized to a resolution of 224×224 with three channels. The facial frame images that have been resized are subsequently provided to our proposed RFE Net.

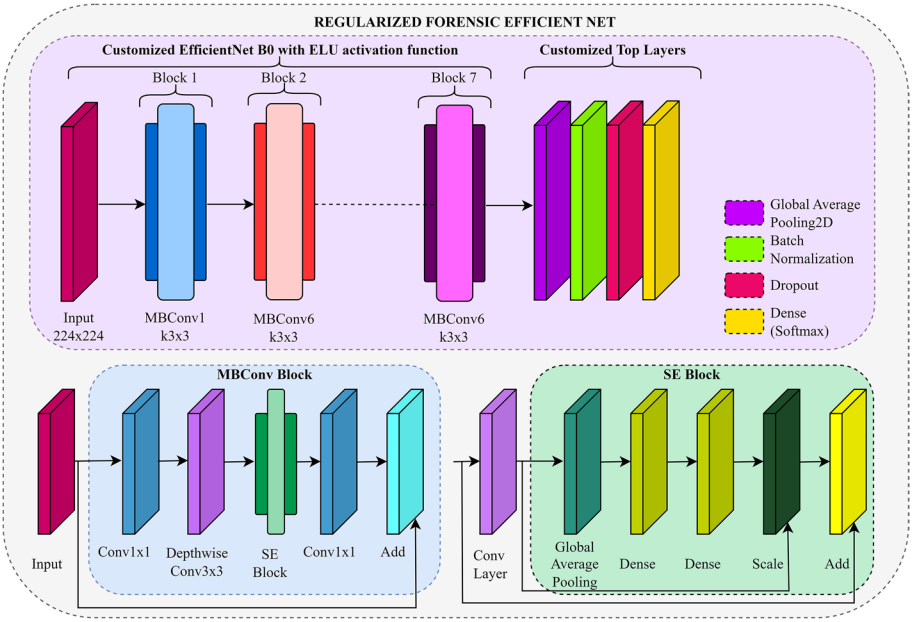

Fig. 2 Proposed RFE NET architecture

3.4 Proposed regularized forensic efficient net

The architecture of the proposed Regularized Forensic Efficient Net for deepfake detection is shown in Fig. 2. In our proposed RFE Net, we altered the activation function, top layers and also introduced several regularization techniques to improve the EfficicentNet-b0 [75], which enhanced performance and efficiency. The use of regularization techniques such as batch normalization, dropout, and early stopping in the proposed RFE Net makes this model resource-efficient and robust. In addition, the use of ELU [76] activation in RFE Net prevents dead neurons, regularizes the gradient flow, and accelerates learning of the model. The architectural design of this model integrates various optimization techniques, including the utilization of depthwise separable convolutions. These convolutions effectively decrease the overall number of parameters and computational expenses, while maintaining the performance of the model. Furthermore, the model integrates efficient residual connections to enhance the flow of gradients and alleviate the problem of vanishing gradients. The parameter count of our RFE Net is 4.01 million, which is lower than that of most CNN models currently available, including EfficientNet (B0-B7), Xception, Inception-v3, InceptionResNet-v2, and DenseNet. Algorithm 2 presents the training process of the proposed model, while the subsequent subsections provide a comprehensive discussion of the specifics of the proposed RFE Net.

Algorithm 1 Two Player zero-sum game

Input: Players P_i with their strategies s_i.

Output: Payoff Matrix with Nash equilibrium.

Initializing Zero-sum game
A finite game, represented as(P_i, Ai, u_i), where:
$P_i \leftarrow$ represents players P_1 and P_2.
$a_i \leftarrow$ actions $a_i = (a_1, \ldots, a_n) \in A$,available for P_i .
$u \leftarrow u_i$ is payoff function created on bases of a_i.
Strategies
Let (P_i, A_i, u_i) be a game for any set Y, let $\prod(Y)$. // the set of all probability distributions.
$Si = \prod(Ai) \leftarrow$For P_i, the set of mixed strategies. // strategy of each P_i changed accordingly.
Nash equilibrium (NE)
while $s_i^* \leftarrow s_i^* \in s_i$ such that $u_i(s_i^*, s_{-i}) \geq u_i(s_i, s_{-i})$ for all strategies $s_i \in S_i$. // The best response for P_i to the strategy profile s_i is a mixed strategy.
NE← optimum response to strategy $s_i = (s_1, \ldots, s_n)$,
if s_i is the optimum response to s_{-i} for all agents i.
end
end
Payoff
$u_i \leftarrow$ utility for P_i of the mixed-strategy profile $s = (s_1, \ldots, s_n)$ is
$u_i(\mathbf{s}) = \sum_{a \in A} u_i(\mathbf{a}) \prod_{j=1}^{n} s_j(a_j)$

$Utility = \frac{a.TP+b.FP+c.FN+d.TN}{TP+FP+TN+FN}$ // utility function for zero-sum game.
Output Maximum of TP and TN, while Minimum of FP and FN.

End

3.4.1 Customized top sequential layers

Sequential model layers are created to pass the features map extracted from the last MBConv block. We customized the model by adding four sequential layers, which are global average pooling, batch normalization, dropout, and dense layer. Batch normalization and dropout layers are used as regularization techniques. The details of the global average pooling and dense sequential layers are as follows:

Global average pooling The Global Average Pooling (GAP) layer is employed to decrease the spatial dimensions of feature maps while retaining crucial information. The process involves computing the mean value across spatial dimensions for every feature map, generating an isolated value for each feature map. This layer diminishes the spatial dimensions of the feature maps, thereby producing a more condensed representation of the salient features. In the final stages of a CNN, this pooling operation frequently replaces fully connected layers, resulting in a model with fewer parameters that is computationally more efficient.

Algorithm 2 Main procedure (Proposed game theory based-RFE NET Training)

RFE Net Hyperparameters: Learning rate α; Epochs E; Sample size N.

Input: P_1**:** Video Repository $V = \{V_1, V_2, V_3, \ldots\ldots\ldots V_n\}$ for each Pristine and Deepfake class.

Output: P_2**:** Detection output, Payoff matrix with maximum *TP* and *TN*.

Initialize: Model = ***RFE Net ()***

Pre-processing:
  For n in N do
  Video facial frame detection through *MTCNN* ().
  Resizing extracted facial frames $F = \{f_1, f_2, f_3, \ldots\ldots\ldots f_n\}$
end
Training process: - RFE Net ():
  For f in F do
    MB Conv block // For deep feature extraction.
      Randomly select a frame F_i from the *P1*.
      ELU // activation function.

$$f(x) = x \; if \; x \geq 0$$

$$f(x) = \alpha * (exp(x) - 1) \; if \; x < 0$$

  Global Average Pooling ← decrease the spatial dimensions of feature maps.
  Batch Normalization ← prevents overfitting and improves the generalization.
  Dropout layer
    Randomly dropout weights.

$$\widehat{w_i} = \begin{cases} w_i, & (p) \\ 0 \end{cases}$$ // (p) is the probability to keep weights w_i and 0 is dropped weight.

  Classification output:
  Dense layer: // last classification layer.
    SoftMax // activation function.

$$softmax(x_i) = \frac{\exp(x_i)}{\sum_{j=1}^{k} \exp(x_j)}$$ // return output for give frame: pristine or deepfake.

  Output: $V = \{V_1, V_2, V_3, \ldots\ldots\ldots V_n\}$ maximum *TP* for Pristine and *TN* for Deepfake.
End

Dense layer In the proposed RFE Net, we used the last fully connected layers with SoftMax activation for the classification of the pristine and deepfake samples. The dense connectivity of neural networks preserves valuable information about the features and weights that are transmitted to subsequent nodes within the hidden layers. The network makes predictions of the class label by identifying the class with the highest probability. This setup allows for a probabilistic interpretation of the predicted class probabilities, making it suitable for binary classification problems.

3.4.2 Feature map generation

The proposed RFE Net model consists of an initial convolutional layer of 224×224 size followed by 7 MBConv blocks. Some of these MBConv blocks consist of subblocks having an expansion layer, a depthwise convolution layer, and a squeeze-and-excitation (SE) block. In the proposed model, the input image gets transmitted through a series of convolutional layers. These layers extract features by employing various filters to detect patterns. The details of the model are given as follows.

MBConv Block This block is a fundamental component of the proposed architecture. It is made up of a few primary elements, which are the expansion layer, depthwise separable convolution, SE block, and skip connection [75]. The depthwise separable convolutional operation involves segregating the spatial and channel-wise convolutions. It consists of a

depthwise convolution followed by a pointwise convolution. Depthwise convolution uses one filter per input channel instead of mixing information across channels and pointwise convolution efficiently combines information across channels using a single convolutional filter. The depthwise convolution performs spatial filtering, while the pointwise convolution performs dimensionality reduction. The SE block is responsible for channel-by-channel feature transformation, enabling the model to prioritize channels with more information. The skip connection connects the input directly to the output, providing a shortcut for gradient flow and enhancing the network's information flow.

SE Block The Squeeze-and-Excitation block enhances the representational capacity of a neural network through the adaptive transformation of channel-wise features. The process includes two distinct operations, namely compression and excitation. The channel descriptor refers to the result obtained from the squeeze operation, wherein the spatial dimensions of the input feature map are combined globally. The excitation operation involves utilizing a series of fully connected layers subsequently followed by an activation function. This process is employed to generate attention weights that are specific to each channel. The attention weights are multiplied elementwise with the input feature map, allowing the network to emphasize informative channels and diminish irrelevant ones.

3.4.3 Activation functions

ELU In each block, we employed the Exponential Linear Unit (ELU) activation function [76] rather than using the traditional Swish activation function to enhance the classification performance and achieve computational efficiency. ELU assigns negative values to negative inputs, allowing the network to capture both positive and negative data. This modification allows for mitigating the dying ReLU issue and improves learning. Implementing ELU [76] rather than Swish enhances the model's generalizability, and its continuity and smoothness can aid in gradient propagation and training optimization. ELU activation function is defined as.

$$\begin{aligned} \mathrm{f(x)} &= \mathrm{x \text{ if } x} \geq 0 \\ \mathrm{f(x)} &= \alpha * (\exp(\mathrm{x}) - 1) \text{ if } \mathrm{x} < 0 \end{aligned} \tag{3}$$

SoftMax In the final layer of the model, the SoftMax activation function is employed for class detection. This function transforms the vector in a probabilistic manner. The SoftMax activation function is utilized to compare the training set with the test set, resulting in a probability distribution that distinguishes between pristine and forged images. SoftMax takes the exponent of each input value divided by the sum of the exponent of all inputs. In the proposed model, the SoftMax function returns 0 for forged and 1 for pristine image, defined as.

$$softmax\ (x_i) = \frac{\exp(\mathrm{x_i})}{\sum_{i=1}^{k} \exp(\mathrm{x_j})} \tag{4}$$

3.5 P_2mixed strategy (s_i) using regularization techniques

In the proposed model, we use several regularization techniques, including batch normalization, dropout, early stopping, and data augmentation. These methods reduce overfitting and

increase model generalization. Batch normalization normalizes inputs inside each mini batch, decreasing internal correlate shift and speeding training. Dropout randomly removes a certain number of units during training, driving the model to learn stronger features. Early stopping terminates training when the model overfits the validation set, preventing performance degradation. Data augmentation artificially expands the training set. The details of these methods are given in subsections.

Batch normalization $(\mathbf{s_i})$ Batch Normalization acts as a regularization, prevents overfitting, and improves the generalization. It normalizes inputs by adjusting the variance and mean of a batch. Normalization accelerates training and reduces internal covariate shifts and enables higher learning rates. It addresses the vanishing gradient problem by ensuring smooth gradient flow.

Let x_i be an input feature and μ and σ^2 represent the mean and variance. The normalization is done as follows:

$$\hat{x}_i = \frac{(x_i - \mu)}{\sqrt{(\sigma^2 + \epsilon)}} \tag{5}$$

Here ε is a small constant for numerical stability. The scaled and shifted output feature is obtained by multiplying the normalized input $\hat{x_i}$ by a scaling factor γ and adding a shifting factor β after normalization.

$$y_i = \gamma \times \hat{x}_i + \beta \tag{6}$$

γ and β are trainable parameters in Batch Normalization. Mini-batch statistics are used to calculate the mean and variance during training. Batch Normalization improves the stability, convergence speed, and generalization ability of neural networks.

Dropout (s_2) We use a dropout layer in the proposed model because it also prevents model from overfitting. This layer randomly drops nodes during model training to achieve an averaged outcome, which encourages the neurons to learn new features for recognition. Once the model is trained, the entire network is used as an inference for the test set. represents output, where (p) is the probability to keep the useful weights, and represents the dropped-out weight.

$$\widehat{w}_i = \begin{cases} w_i, \\ 0 \end{cases} \tag{7}$$

Early stopping (s_3) To save training time for our model we used an early stopping strategy [77]. It monitors the performance of the model and effectively reduces the duration of the training process. Early stopping measures performance by setting the parameters for the performance measure, in this case, validation loss or accuracy. Other parameters include minimum or maximum mode and the verbose value.

$$\text{es} = \text{EarlyStopping}\big(\text{monitor} =' \text{val}'_{\text{loss}}, \text{mode} =' min', \text{verbose} = 1\big) \tag{8}$$

Data augmentation (s_4) To resolve the issue of class imbalance in several datasets like DFDC, we augment the training set. These augmentations [78] include vertical and horizontal flips, rotation, cropping, and horizontal and vertical shifts. Moreover, we used salt and pepper noise, and median blurring for several test set experiments to evaluate the efficacy of the RFE Net.

4 Experimental results and discussion

This section provides a comprehensive overview of the various experiments conducted to evaluate the efficacy of the proposed RFE Net. A comprehensive analysis of performance evaluations involving open and closed sets is provided. Additionally, different experiments like cross-set, cross-corpus, and post-processing attacks are conducted to evaluate the generalizability of the proposed method across different scenarios. Finally, a comparative analysis is conducted to compare the RFE Net with contemporary methods. Details of datasets used for evaluation are also presented in this section. Moreover, we employed the accuracy, area under the curve (AUC), and precision-recall (PR) curve for the performance evaluation of our method.

4.1 Datasets

Four standard datasets were chosen to assess the effectiveness of the proposed model. The following section provides a comprehensive discussion of the details of all datasets.

4.1.1 Face forensic ++

The Face Forensics++dataset [20] is a benchmark dataset composed of pristine and two major forgery subsets: (i) identity swap manipulation and (ii) expression swap. The identity swap manipulation subset contains face swap, face shifter, and deepfakes subsets, and expression swap consists of face2face, and neural textures as shown in Fig. 3. For forgery fabrication, 1000 pristine videos are collected from YouTube and then manipulated using a fully automated GANs. In the face swap subset, the source facial portion is replaced with the target one using the landmark positions such as nose, eyes, and mouth. In face2face, the facial expressions of the target are substituted with the source video using the facial re-enactment method. In the deepfake subset, deep learning modification techniques based on auto-encoders are used to manipulate the faces of the source and target actors. Neural textures are created using a fully automatic face rendering technique. The face shifter subset was later added to the FF++dataset in which GAN was employed to manipulate the faces in the target videos. In FF++, each video contains distinctive faces of people from

Fig. 3 FF++dataset sample (Left to right) Real, Faceswap, DeepFakes, Face 2 Face, Neural Textures and Face Shifter

Fig. 4 Samples of the DFDC dataset. Top row: Real Frames, Bottom row: Fake Frames

different ethnicities, with facial accessories like spectacles or facial hair, and frames with contrasting illumination or occlusion conditions. For our experiments, we used the light-compressed, lossless, c23 version of FF++.

4.1.2 DFDC

The DFDC preview dataset [79] consists of 5,000 real and deepfake videos. GAN and non-learned techniques are used to create fake videos, while actual ones were recordings of hired actors. Using facial manipulation techniques such as DeepFakes and Face2Face, fake recordings are created. Indoor and outdoor settings, day and night lighting, subject proximity to the camera, posing changes, and other factors are all taken into attention in dataset creation. DFDC is diverse in terms of skin tone, gender, age, etc. The dataset concentrates on advanced deepfakes and demonstrates the difficulties of deepfake technology. The samples from the DFDC dataset are given in Fig. 4.

4.1.3 World Leader Dataset (WLDR)

World Leader Dataset (WLDR) [31] contains YouTube videos of US leaders such as Hillary Clinton, Barack Obama, Bernie Sanders, Elizabeth Warren, and Donald Trump. The dataset consists of clips of the leaders, comedic impersonators, and faceswap subsets. Original, comedic impersonators and face swaps of different leaders can be seen in Fig. 5. Additionally, the Obama subset contains lip-sync and puppet-master forgeries as well. More recently, Joe Biden's comedic impersonator and faceswap were added to the dataset. Each video was captured by focusing on points of interest (leader), addressing the speech's informal surroundings. The full-frontal faces are captured on a static camera; however, some shots are captured with a slow zooming technique with the leader speaking throughout the 30 frames per second video clip. The WLDR faceswap subset was created with GANs by swapping the original face of a leader with that of an impersonator. This dataset has a class imbalance issue.

Fig. 5 Samples of the WLDR dataset. Top to bottom is the Real, comedic impersonator, and Faceswap images of different US leaders

Fig. 6 Samples of the CelebDF dataset. Top row: Real Frames, Bottom row: Fake Frames

4.1.4 CelebDF

The Celeb-DF dataset [80] comprises 590 genuine and 5639 deepfake videos. The Authentic YouTube videos contain interviews with celebrities of various genders, ages, and ethnic backgrounds, and illumination conditions of real-world videos display a wide variety. To reduce the contrast between the altered areas, the deepfake creation approach improves the brightness and contrast of facial images. As a result, altered videos with higher visual quality are more deceptive. Figure 6 illustrates a few dataset samples.

Fig. 7 Samples of the DFFD dataset. Real Sample, PGGAN, STARGAN, StyleGAN-CelebA, StyleGAN-FFHQ

4.1.5 DFFD

The Diverse Fake Face Dataset (DFFD) [81] provides a more extensive selection of fake face types than previous datasets. This is crucial for the accurate detection and localization of facial manipulations. There are four primary categories of facial manipulations in the dataset: identity swap, expression swap, attribute manipulation, and entirely synthesized faces, which are produced using cutting-edge techniques. It encompasses a balanced distribution of gender, age, and face measurement among 47.7% male and 52.3% female subjects, with a primary age range of 21 to 50. The FFHQ, CelebA, and FF++datasets were used to source real face images, which exhibit a wide range of variations in terms of gender, age, ethnicity, and other factors. Data from FF++and other sources were employed for identity and expression exchanges. FaceAPP and StarGAN were employed to manipulate attributes, resulting in the generation of 92,000 images. Using pre-trained models of PGGAN and StyleGAN, the entire visage was synthesized, resulting in the production of 300,000 high-quality fake images. Consequently, the DFFD provides a comprehensive dataset to facilitate the advancement of facial manipulation detection. The samples of DFFD are shown in Fig. 7.

4.2 Experimentation protocol

Every individual frame of all the datasets was extracted using the MTCNN. During face frame extraction, a parameter was used to resize each frame to 224×224 pixels. For each experiment, 80% of the frames were used for training, while the remaining 20% were used for the test set. The parameters of the model were set as follows: learning rate = 0.0001, L2, batch size = 100, and epochs = 25. The model was trained using the Adam optimizer and the binary cross-entropy loss. Experiments were conducted on high-performance computers with GPU nodes that met the following requirements: 4 NVIDIA Tesla V100 16G GPUs with NV Link, 192 GB RAM, and 48 CPU Cores at 2.10 GHz.

4.3 Evaluation measures

The following evaluation measures were used to measure the performance of the proposed technique. For these measures, the model was evaluated on all datasets.

4.3.1 Accuracy

Accuracy is used to measure the percentage of accurately classified members of the given classes. It can be defined as the sum of the TN and TP rates divided by the total number of samples.

4.3.2 Area under the curve

The AUC represents a graphical representation of the relationship between the True Positive Rate (TPR) and the False Positive Rate (FPR). TPR and FPR are computed at various thresholds, with optimal AUC values corresponding to higher TPR and lower FPR.

4.3.3 Precision/Recall curve

The provided visual representation depicts the combination of Precision and Recall, also known as True Positive Rate (TPR). A model that exhibits a higher PR curve closer to the y-axis is indicative of superior performance.

4.4 RFE net analysis and ablation study

To develop a robust and generalized deepfake detector we perform several experiments, which include regularization and activation function selection. This experiment also includes the analysis based on regularization techniques used in the proposed approach for detecting deepfakes.

4.4.1 Performance evaluation of the proposed detector (P_2) based on regularization techniques

This experiment aims to select best regularization techniques $\left(s_i\right)$ and evaluate the efficacy of the model with and without regularization techniques. During training, dropout regularization reduces the model's reliance on particular neurons, thereby enhancing its generalizability. In addition, data augmentation was utilized to increase the effective size of our limited dataset, thereby enhancing the adaptability of the model. The early stopping method contributed to a simpler optimization process during training, resulting in a shorter training period [77]. Overall selecting a regularization strategy reduces overfitting and preserves key features in the data.

Regularization selection We select dropout (s_1) and L2 regularization (s_2) to prevent overfitting, we also add a batch normalization layer (s_3) which boosts regularization and model performance. Batch normalization normalizes layer activations, stabilizing the model and decreasing weight initializations. L2 regularization adds a penalty term to the loss function to reduce model complexity by selecting lower weights. We attain an accuracy of 96.5% on using L2 and batch normalization with our base model. However, dropout randomly sets a fraction of input units to zero during training, inhibiting neuron co-adaptation and improving generalization. Dropout has a larger regularization effect than L2 regularization, as we attain the highest accuracy of 98.8% on combined FF++by a combination of dropout and batch normalization. In comparison to L2, we finalize dropout and batch normalization as

the best regularization techniques $\left(s_i^*\right)$. This combination leads the model to even better generalization and improved training speed.

Dropout layer An experiment was created to compare the performance of the proposed RFE Net with and without dropout regularization as a strategy to measure the training and validation loss. Before applying dropout regularization, the model was trained on the combined FF++dataset. During the training process, the validation loss decreases slightly and then rises again due to model overfitting. When the detector strategy was modified by adding a dropout layer, it regularized the weight by decreasing the validation and training loss. From this experiment, it can be inferred that dropout regularization helps to prevent overfitting.

Early stopping To better investigate the computational cost of the proposed RFE Net, an early stopping regularization strategy was used to prevent a model from unnecessary training. To train the model without using an early stopping strategy, the training epochs were set to 50 in the first experiment. This model generates accurate training set results but is less accurate on the validation set. Changing this strategy, early stopping was introduced. Performance measures were configured to monitor the "validation loss" and the mode was set to "min" to minimize the validation loss. The "verbose" parameter was used to show training progress. This parameter also validated each of the epochs and stopped our model from unnecessary training. This model produced the best training and validation accuracy only on 7 epochs. The early stopping regularization strategy (s_3) made the model better fit the training sample, prevented the model from iterating uselessly, and provided a significant reduction in computational cost.

4.4.2 Comparison with deep learning models

This experiment was conducted to compare the effectiveness of our proposed RFE Net and other deep learning architectures with and without regularization. To test the performance of our model, we utilize the combined FF++dataset. This evaluation involved a comparative analysis of our model against others such as ResNet, VGG16, Inception, and DenseNet. All the models, including the proposed model, are trained using the same dataset and the game scenarios. The proposed method, even without regularization, demonstrates enhanced accuracy while utilizing a smaller number of parameters, thereby increasing its efficiency as well. In addition, we train all models with regularization. The accuracy of each model increased with parameter decreases, showing the effectiveness

Table 1 Comparison with different deep-learning models

Deep learning Models	Without Regularization		With Regularization	
	Accuracy (%)	Parameters	Accuracy (%)	Parameters
ResNet50	85.6	25.6 M	88.5	23.4 M
VGG16	87.2	138.4 M	92.3	120.5 M
InceptionV3	90.7	23.9 M	94.5	22.0 M
DenseNet121	92.4	8.1 M	95.5	7.3 M
Proposed RFE Net	**97.3**	**5.3 M**	**98.8**	**4.0 M**

Table 2 Performance evaluation with different activation functions

Activation Functions	Accuracy (%)	Training time
Swish	98.53	2 h
ELU	**99.39**	**1 h 40 min**
GELU	97.07	3 h
SELU	97.78	2 h 30 min
RELU	98.71	2 h

of regularization. In comparison with others, the proposed model balances depth, width, and resolution in layers, enabling it to maintain efficiency while delivering outperforming results. Table 1 shows the comparison of the findings in terms of accuracy and parameter information. The results demonstrate that our RFE NET outperforms all other deep-learning models while using the fewest number of parameters.

4.4.3 Comparison of different activation functions

In this experiment, we investigate the effect of various activation functions on the proposed RFE Net. The model was tested using several activation functions, and the results are shown in Table 2. The experiment was conducted using the combined FF++dataset, with the same experimental protocols defined in Section 4.4. Exponential Linear Unit (ELU) was found to be the most promising activation function among all, exhibiting superior performance on all datasets. One of the primary benefits of the ELU is its ability to mitigate the issue of.

the dying ReLU by providing the formation of the negative slope during the training process [76]. This feature prevents neurons from becoming inactive, thereby enhancing the network's capacity to effectively learn complex functions. Moreover, the inherent smoothness of ELU contributes to its computational efficiency, rendering it highly effective even in situations with limited resources. This characteristic makes ELU a compelling choice for an activation function in our model.

4.5 Performance evaluation of the proposed RFE Net

To test the robustness of the proposed RFE Net for deepfake detection, rigorous experiments were designed using mixed strategies (s_i)of both players P_1 and P_2, on the FF++, WLDR, CelebDF and DFDC datasets. P_1 strategies include dataset formation and combination, whereas P_2 in response uses its best strategy to detect between given classes.

Table 3 Performance evaluation of RFE Net on FF++

Generative Technique	Faceswap	Deepfakes	Face2Face	Face Shifter	Neural Textures	Combined
Acc (%)	99.3	98.4	98.1	97.9	92.5	98.8
PR	0.98	0.97	0.98	0.97	0.95	0.98
AUC	1.00	0.98	0.98	0.98	0.90	0.95

*(Acc = Accuracy, PR = Precision, AUC = Area under curve)

4.5.1 Evaluation on FF++

For the FF++dataset, two different dataset division strategies were employed, as follows:

Strategy 1(s_1) Pristine video frames were used to test the model against each given fake class: face swap, deepfake, face2face, neural texture, and face shifter. Overall, superior results were attained for all sets, but for the deepfake set, the highest accuracy of 99.36% and an AUC of 1.00 was achieved. The results of the proposed RFE Net with all subsets of FF++are presented in Table 3.

Strategy 2 (s_2) To test the generalizability of the proposed model, the P_1 strategy was changed by combining all the given fake classes into one combined class. Combined fake and pristine samples were used to train the RFE Net. In this experiment, the highest accuracy was 98% and an AUC of 1.00 was observed. The results of the combined set are provided in Table 3.

The results presented in Table 3 show that the RFE Net (P_2) is remarkably effective at identifying all types of manipulated subsets (P_1). This observation demonstrates the model's robust ability to detect accurately manipulated features generated using deep learning techniques. The aforementioned results indicate that the proposed model (P_2) effectively incorporates the essential characteristics required for detecting alterations. Still, it can be noted that the performance of the proposed model exhibits a slightly low accuracy on the neural texture dataset achieving scores of 92.5%. This set uses expression-swapping techniques to create counterfeit features with minimal changes, making them extremely difficult to identify. Therefore, identifying this particular form of manipulation presents a difficult task for the model.

4.5.2 Evaluation on WLDR

For WLDR, the model was tested on pristine videos of each leader with the corresponding comedic impersonator and face swap. In this experiment, three different dataset division strategies were implemented from P_1 and in response P_2 is also trained against all the dataset manipulation strategies given below. The results of all strategies are shown in Table 4.

Table 4 Performance evaluation of RFE Net on WLDR

Leaders	Obama			JB		Clinton		Warren		Sander		Trump	Combine	
Generative Technique	FS	IM	Lip	FS	IM	FS	IM	FS	IM	FS	IM	IM	IM	FS
Acc (%)	99.1	99.6	97.9	99.3	100	99.2	97.8	92.0	95.8	95.5	99.3	99.0	99.8	98.7
PR	0.98	0.97	0.99	1.00	1.00	1.00	0.99	0.96	1.00	0.99	1.00	1.00	0.99	0.98
AUC	0.99	0.98	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.99	1.00	1.00	0.99	0.99

*(FS = Faceswap, IM = Impersonator, Lip = Lip-sync)

Strategy 1 (s_1) In this experiment, the RFE Net was trained on each leader's original class, the corresponding comedic impersonator, and faceswap. The Obama videos were additionally trained on the lip-sync class. On WLDR, good accuracy was achieved overall for all the leaders shown in Table 4, however, for Obama, Clinton, Sander, Trump (impersonator), and Obama, bidden (faceswap), the highest accuracy of 99% and AUC up to 1.00 was observed.

Strategy 2 (s_2) In this experiment, pristine/original samples of all leaders were combined in one class, face swap in another class, and both were used to train the proposed RFE Net. Experimental results show that the RFE Net produced a remarkable accuracy of 98% and 0.99 AUC on the combined face swap class.

Strategy 3 (s_3) In this experiment, the pristine samples of all leaders constituted one class, and comedic impersonators in another. The model was trained in these classes, and in this experiment, the model attained 99% accuracy and 0.99 AUC on the combined impersonator class.

Table 4 demonstrates that the model (P_2) performed exceptionally well on all categories of leader deepfakes. In many subsets, the model can obtain the highest AUC and accuracy, except for Warren, with 92% accuracy. The dataset reveals that Warren's faceswap class was similar to its real class, which increases the likelihood of fewer false positives in comparison to other leaders. The findings from the analysis demonstrate that the proposed model exhibits the ability to differentiate between authentic and fake samples based on their distinctive attributes.

4.5.3 4.5.3 Evaluation on DFFD

For the DFFD dataset, two different dataset division strategies were employed, as follows:

Strategy 1(s_1) Real video frames were used to test the model against each given fake class: PGGAN, STARGAN, StyleGAN-CelebA, StyleGAN-FFHQ. Overall, superior results were attained for all sets, but for the StyleGAN-FFHQ set, the highest accuracy of 99.99% and an AUC of 1.00 was achieved. The results of the proposed RFE Net with all subsets of FF++ are presented in Table 5.

Table 5 Performance evaluation of RFE Net on DFFD

Generative Technique	PGGAN	STARGAN	StyleGAN-CelebA	StyleGAN-FFHQ	Combined
Acc (%)	99.5	99.3	99.4	99.9	99.6
PR	0.97	0.91	0.99	0.96	0.99
AUC	0.99	0.97	0.98	1.00	0.97

*(Acc = Accuracy, PR = Precision, AUC = Area under curve)

Strategy 2 (s_2) To evaluate the generalizability of the proposed model, the P1 strategy was modified by combining all the provided fake classes into a single combined class. The RFE Net was trained using a combination of pristine and fake samples. An AUC of 0.97 was observed in this experiment, with a maximum accuracy of 98%. Table 5 contains the outcomes of the combined set.

The RFE Net (P2) is remarkably effective at identifying all varieties of manipulated subsets, as demonstrated by the results in Table 5. This observation illustrates the model's capacity to precisely identify manipulated features that are produced through the different GANs. The aforementioned results suggest that the proposed model (P2) effectively integrates the fundamental characteristics necessary for the detection of alterations. However, it is important to acknowledge that the proposed model demonstrates a high level of accuracy on all DFFD subsets.

4.5.4 Evaluation on Celeb-DF

In this experiment, the model was evaluated on pristine videos with the deepfake with two strategies.

Strategy 1 (s_1) To assess the effectiveness of the deepfake detection model, a comparative analysis was conducted between the original and synthetic samples of CelebDF. Our RFE Net attained a notable accuracy of 91.9% and 0.93 AUC. Despite the data imbalance issue as this dataset contains substantially more fake videos (5639) than real videos (590), the proposed method was able to identify key characteristics based on color and texture.

Strategy 2 (s_2) In this experiment, we employed several augmentation techniques through CLoDSA [78] to create a balanced dataset. Augmentation was applied to a genuine class to create several samples equivalent to those of a fake class. Several transformations were applied to the real data, including horizontal and vertical flipping, rotation, luminance adjustment, and cropping. In comparison, augmentation enhanced the results by 94.3% and resulted in an AUC of 0.93.

Even though Celeb-DF is a highly imbalanced dataset, the proposed method effectively captures the characteristics of the samples by analyzing changes in color and texture. Remarkably, it distinguishes highly realistic swapped features within the Celeb-DF dataset with minimal color variance and reduced temporal flickering. After applying augmentation to this dataset, the results improved and demonstrated the method's (P_2) excellent performance.

4.5.5 Evaluation on DFDC

In this experiment, the model was tested on pristine videos with the deepfake using two strategies, one with an imbalanced dataset and the second with a balanced one.

Strategy 1 (s_1) To evaluate the effectiveness of the proposed model in identifying deepfakes within the DFDC dataset, both genuine and fake samples were analyzed. RFE Net

attained an accuracy rate of 89.9% and an AUC value of 0.80. Upon examination of the DFDC dataset, one major reason is a class imbalance from 5k videos around 28% of samples are real, and the remaining is deepfake generated. Furthermore, many videos were recorded in conditions of dim lighting, posing a challenge to the model's ability to differentiate visual components. Significantly, the proposed model can differentiate between authentic and fake samples, even under conditions of limited lighting and side poses. Despite the diversity of the dataset and these variations, the proposed model detects deepfake artifacts with remarkable accuracy.

Strategy 2 (s_2) DFDC dataset is highly imbalanced, with the fake class having more samples than the real one. To address this issue, several augmentation techniques were utilized, such as horizontal and vertical flipping, rotation, luminance adjustment, and cropping. These methods helped to increase the number of samples in the real class to match that of the fake class, creating a more balanced dataset. As a result of this augmentation, the accuracy and AUC reported on DFDC were improved by 90.3% and 0.90 compared to the original dataset. This indicates that augmentation is an effective way to enhance the performance of a model.

Most of the videos in the DFDC dataset were captured under extremely dim illumination conditions, which created a challenge to the discrimination abilities of the model. Remarkably, our model P_2 can identify pristine samples from deepfake samples even in dim lighting and side poses. Most of the videos in the dataset feature two actors conversing from side angles. Despite encountering such a wide variety of dataset variations, the proposed model remains to detect deepfake anomalies with remarkable precision. In addition, training the model on the augmented dataset also improved its accuracy.

Overall, these results on all datasets show that RFE Net performs remarkably well for classification. It should be noted that the player of interest P_2 performs well even when employing different strategies for P_1. More specifically, the use of regularization techniques as a game-theoretic strategy for P_2 enables the model to learn the features of the original and deepfaked samples rather than memorize them. The proposed strategies not only generate the highest accuracy but also produce higher PR and AUC.

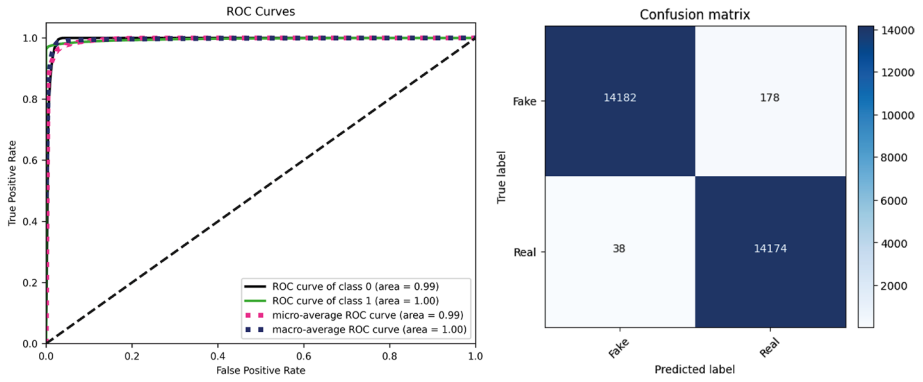

Fig. 8 Payoff matrix and AUC of proposed model

4.6 Payoff matrix analysis

To better investigate the "True positive rate" and "True negative rate" scenarios concerned with the different strategies of P_1 and P_2, a payoff matrix was used for analysis. This experiment was designed to measure the performance of the game theoretic RFE Net on all datasets. As the payoff matrix shown in Fig. 1, NE is presented as the maximum of TP and TN. As seen in Fig. 8, the best payoff matrix representation was achieved on the faceswap subset of FF++datasets. On the FF++dataset, RFE Net achieved the highest TP and TN, 14,182 pristine and 14,174 deepfake samples, and achieved an overall TP and TN of 99%. The incorrectly classified 178 deepfake and 38 pristine samples represent a minimal FP and FN. These maximum TP and TN represent that the game achieved an NE state using pessimistic approaches, which lead our game to an overall maximum payoff. This analysis shows the robustness of our RFE Net for the detection of deepfake using game theory.

4.7 AUC analysis

To determine the responsiveness of the proposed game-theoretic RFE Net, the receiver operating characteristics (ROC) curve was created. The ROC curve measures the trade-off between higher TPR and lower FPR. The AUC measures the model's ability to discriminate between pristine and deepfake samples. From the ROC curves in Fig. 8, it can be observed that the proposed method attained exceptional results. The black line in Fig. 8 represents the curve for the pristine class and the green line is the curve for deepfakes. The results where these lines are more on the top-left side of the graph present the best AUC. Tables illustrate the AUCs of the proposed method on all datasets. The higher AUC shows that the model is discriminating well between the pristine and deepfake videos.

4.8 Performance evaluation of RFE Net in cross-set scenarios

To evaluate the generalizability of the RFE Net, rigorous cross-set experiments were designed to evaluate the performance of a proposed model on FF++and WLDR datasets using different strategies. This experiment is applicable to the mentioned datasets due to additional subsets within these datasets.

4.8.1 Evaluation on FF++

Strategy 1 (s_1) This experiment was designed to evaluate the model's robustness for identity swap and expression swap subsets of the FF++dataset. To identify instances of identity swapping, the authentic and fake samples were used to train RFE Net from combined sets such as faceswap, deepfakes, and face shifter (FS+DF+SH). Face2Face and neural texture datasets (F2F+NT) were combined to train the model in the context of expression swapping. The close-set classification (CSC) was performed on the trained models. In the context of close-set testing, the model was subjected to individual testing on each of its subsets, regardless of whether it was trained on identity swap or expression swap sets. In this experiment, our model demonstrated superior performance on each CSC set. The corresponding results can be found in Table 6.

Table 6 Cross-set evaluation of proposed RFE Net on FF++

Identity Swap	Train Set	FS + DF + SH				
		CSC			OSC	
	Test Set	FS	DF	SH	NT	F2F
	Acc (%)	97.6	97.8	96.6	65.4	72.6
	PR	0.96	0.96	0.96	0.65	0.74
	AUC	0.96	0.97	0.95	0.65	0.75
Expression Swap	Train Set	F2F + NT				
		CSC		OSC		
	Test Set	F2F	NT	FS	DF	SH
	Acc (%)	92.6	90.2	70.4	66.7	53.3
	PR	0.90	0.90	0.72	0.65	0.55
	AUC	0.85	0.90	0.73	0. 67	0. 54

*(FS = Faceswap, DF = deepfake, F2F = Face2Face, NT = Neural Textures, SH = Shifter)

Strategy 2 (s_2) In the second experiment, we decided to change the strategy and make an open-set classification (OSC). This allowed us to test identity swapping on the expression-swapping subsets and vice versa. For this experiment, we trained our model on (FS + DF + SH) and then tested it on the subsets of F2F and NT. It can be observed that the CSC attained better results because the model was trained on close-set train sets. From this cross-set evaluation on FF++, the Strategy is considered the best strategy which is followed by the detector for deepfake detection.

4.8.2 Evaluation on WLDR

In this experiment, intra-class cross-set experiments were performed for the WLDR dataset.

Strategy 1 (s_1) In this evaluation, each leader's comedic impersonator was tested against the same leader's face swap and vice versa. The motivation of this experiment was to check how well our models discriminate between an impersonator of a leader from its deepfake. The cross-set results of an impersonator on the WLDR dataset are mentioned in Table 7. These results show the robustness of the proposed model on the WLDR dataset as it successfully discriminates between the intra-class forgeries even in class imbalance scenarios. In this context, our model's overall performance is convincing because it generalizes well to cross-set samples.

Table 7 Cross-set evaluation of proposed RFE Net on WLDR

Leaders	Obama		Clinton		Sander		Warren	
Test Set	Imp	FS	Imp	Imp	FS	FS	Imp	FS
Train Set	FS	Imp	FS	FS	Imp	Imp	FS	Imp
Acc (%)	93.64	95.86	96.13	85.21	80.77	96.46	91.60	91.73
PR	0.94	0. 95	0.95	0. 85	0. 81	0. 95	0. 90	0. 90
AUC	0.92	0. 93	0.95	0. 84	0. 80	0. 95	0. 90	0. 90

4.8.3 Cross-set comparison with contemporary methods

In comparison with our highest cross-set accuracy of 97% on the FF++dataset, the existing methods [82] achieved between 60% and 65% cross-set accuracy; [57] attained a high accuracy of 75.30%; and [58] achieved an accuracy of 68%. This experiment demonstrated the superior performance of our model in close-set scenarios than in open-set scenarios on the FF++dataset. It indicates our model's ability to detect fake traits that differ from those detected in the training data. The results for open-set testing were good, except for the face shifter, which is generated by the GAN-based technique. The intricate textures make it difficult to reliably capture the distinguishing characteristics, resulting in lower performance for the model. Despite this, the proposed method demonstrated outstanding generalizability in CSC testing and reasonable performance in the OSC scenario. For the WLDR dataset, all experiments attained outstanding results, which demonstrate the generalizability of our RFE Net on cross-set experiments. The overall results indicate that our method is more generalizable in comparison to existing techniques.

4.9 Cross-corpora evaluation

To check the inter-class generalizability of the proposed method, we designed a cross-corpora experiment. Cross-dataset experiments were devised on all datasets to evaluate the generalizability of RFE Net among totally different datasets.

Strategy 1 (s_1) In this experiment, we combined all leaders' face swap subsets. As the combined subset contained a smaller number of videos, we used augmentation to increase the number of samples. We applied vertical and horizontal flips, rotation, cropping, and horizontal and vertical augmentation techniques using the data augmentation library CLoDSA [78]. The combined WLDR is trained, and model performance is evaluated with FF++, CelebDF, and DFDC datasets.

Strategy 2 (s_2) For this experiment, identity-swapping subsets of the FF++dataset are combined to evaluate with WLDR, CelebDF, and DFDC datasets. In this experiment, less accurate results were achieved because the WLDR dataset is not as diverse as FF++. But still, the model precisely detects the TP (Pristine) sample available in the FF++dataset. We trained our model with a combined train set and evaluated it with WLDR for each leader's face swap subset. In this experiment on the Clinton subset, we attained the highest accuracy of 73.4%, because the model is trained on diverse identity-swapped samples. Even in diverse case cross-corpora evaluation, the model achieved more than 73% accuracy, which is far better than contemporary methods.

Strategy 3 (s_3) This experiment involved the utilization of a trained model on the CelebDF, which was subsequently evaluated using the DFDC, WLDR, and FF++test set. The WLDR test set includes the collection of faceswap test sets of all leaders, while the FF++dataset's test set comprises samples generated using the identity-swapping technique.

Strategy 4 (s_4) The last cross-corpora experiment involved training the RFE NET model on the DFDC dataset and subsequently evaluating its performance using the CelebDF, WLDR, and FF++datasets. A test set was created for the WLDR and FF++models using the technique as used in .

Table 8 Cross-corpora evaluation of proposed RFE Net

Train Set	FF++				WLRD				Celeb-DF				DFDC				DFFD			
Test Set	WLRD	DFDC	Celeb -DF	DFFD	Celeb-DF	DFDC	DFFD	FF++	Celeb-DF	DFDC	FF++	DFFD	WLRD	DFDC	FF++	DFFD	Celeb -DF	WLRD	FF++	DFDC
Acc %	68.0	70.3	67.3	70.2	76.6	70.7	75.9	69.1	76.6	70.7	69.1	72.3	68.5	63.9	65.8	67.6	78.1	67.2	70.3	69.8
AUC	0.70	0.73	0.68	0.70	0.77	0.70	0.76	0.57	0.77	0.70	0.57	0.72	0.67	0.62	0.65	0.66	0.77	0.65	0.68	0.71

4.9.1 Cross-corpora comparison with contemporary methods

This experiment is conducted to demonstrate the effectiveness of a cross-corpora evaluation of our technique. Comparing the cross-corpora evaluation with the existing method [57, 58, 60], our method attains good cross-corpora results. The method [58] achieved cross-dataset results between 40 and 60%, whereas for [60], the best cross-corpora value is 76.1%. The highest cross-corpora accuracy attained by our model is 78.19%. It is shown in Table 8 that DFDC and CelebDF, DFFD outperformed the other datasets when compared to FF++ and WLRD. Compared to other datasets, Celeb-DF, DFFD, and DFDC comprise a variety of samples with different illumination conditions, backgrounds, ages, genders, and ethnicities and are created through face-swapping techniques.

In the cross-dataset scenario, our model achieved remarkable accuracy on these datasets. In contrast, the FF++ dataset includes techniques for expression and identity-swapping. Moreover, WLRD is restricted to videos of only five leaders, so it cannot produce convincing results when compared to other datasets. Therefore, the efficacy of these datasets was inferior to that of DFDC, DFFD, and Celeb-DF. Given the diversity of each dataset, it is reasonable to expect cross-dataset results between 60 and 70%. Each dataset varies due to the inclusion of different ethnicities, ages, geographic locations, lighting conditions, and face accessories.

4.10 Generalizability analysis on multiple post-processing attacks

To determine the generalizability and robustness of the proposed model in the context of various adversarial attacks including noise, blur, pixel dropout, and elastic deformation, the P_1 data manipulator acts as the attacker. It is essential to note that the proposed model P_2 was not trained for such attacks. We conducted this experiment on all datasets and details are provided in this section.

To test the efficiency of our RFE Net on unseen attacked instances, we devised an experiment that used four different strategies for attacking the test set images of all datasets. It is important to highlight that the models were only trained on normal data, attacked samples were not included in the training. These techniques include ($\mathbf{s}_1$): Salt and pepper noise, ($\mathbf{s}_2$): Gaussian blur (kernel = 5), ($\mathbf{s}_3$): Pixel Dropout (0.05%), and ($\mathbf{s}_4$): Elastic deformation ($\alpha = 5$, sigma = 0.05).

For each FF++, DFFD, and WLDR dataset, we merged all subsets into one training set, whereas the CelebDF and DFDC datasets were used as a whole. Table 9 shows the results of all post-processing attacks on datasets. Overall, we obtained accurate outcomes for all

Table 9 Performance evaluation of proposed model on post-processing attacks

Dataset	Accuracy (%)			
	Blur	Noise	Pixel Dropout	Elastic Deformation
FF++	92.5	85.4	78.3	72.6
WLDR	94.5	89.5	80.2	76.9
DFFD	93.2	86.1	79.4	74.5
CelebDF	90.4	86.7	82.5	70.3
DFDC	83.7	74.6	71.9	69.2

datasets, including attacked samples, except for elastic deformation-attacked instances. Elastic deformation simulates model-deceiving distortions by applying non-rigid changes to the input image. Comparing the results of attacked images to those of normal images, Table 9 demonstrates a decrease in the detection accuracy for the attacked samples. RFE Net classifies attacked samples with an accuracy of at least 69.2% for the DFDC dataset. Analyzing the DFDC, it was noticed that most of the videos were captured in extremely dim illumination, making it challenging for the model to differentiate between fake and genuine. In addition, the use of elastic deformation makes it more challenging for a model. Still, our model is reasonably resistant to post-processing attacks due to the use of dropout regularization and data augmentation. The detector's improved resistance may prevent attacks and strengthen the defensive approach as a stable, mutually optimum response by minimizing exploit attempt returns. Also implied is that RFE Net outperforms on both attacked and non-attacked samples. Because the generalizability of the proposed RFE Net enables it to achieve such satisfactory detection results for the attacked images, this allows the model to become resistant to unseen samples not observed during the training.

4.11 Explain ability analysis through heat maps

Deep learning models are considered black box models that make classification decisions for given classes based on deep features without showing any visual evidence. For explainability purposes, heatmaps improve the visualization of what the network has learned during deep feature extraction. To visualize the explainability factor of our RFE Net, we created heatmaps through the Grad-cam [83] approach. Selecting the Grad-cam in comparison to other methods like saliency maps, lime, and guided backpropagation offers a more precise and interpretable visualization of the model's attention. Grad-cam uses gradients in the final convolutional layer, which not only helps in interpreting the model's decision-making process but also enhances the transparency of the model. Figure 9 depicts the heatmaps matching the proposed RFE NET's final convolution layer overlaid with several types of deepfakes. The RFE NET focuses on the silhouette in the face region where the manipulation is present. The visual study supports our claim that the RFE NET is suitable for deepfakes detection. It is clear from the visual study that the RFE NET emphasizes the regions inside the face, like the foreheads, brows, nose, eyes, etc. The focus of heatmaps on distinct sections of the facial region indicates that our RFE NET model is focusing on relevant sections in the frame. In comparison, we generated heatmaps after post-processing attacks it can be seen in Fig. 9 that noise distorts pixel values, resulting in incorrect focus areas in heatmaps. Blurring reduces detail and smooths pixels, resulting in diffused activations in heatmaps as the model attempts to detect subtle features. The model's inability to deal with incomplete inputs is demonstrated by the data gaps that are created by dropping pixels. These gaps are visible in heatmaps as areas of reduced or absent activation. Elastic Deformation causes image distortion that misaligns features. However, our model is still resistant to post-processing attacks due to the use of dropout regularization and data augmentation.

4.12 Comparison with contemporary methods

To analyze the effectiveness of our game-theoretic RFE NET for deepfake detection, it was compared with the existing state-of-the-art methods. We compared our model's performance on the FF++dataset using [20, 38, 39, 51, 56–60], on the DFDC dataset using [56, 59, 60], on the DFFD using [57], on the CelebDF dataset using [38, 39, 60], and

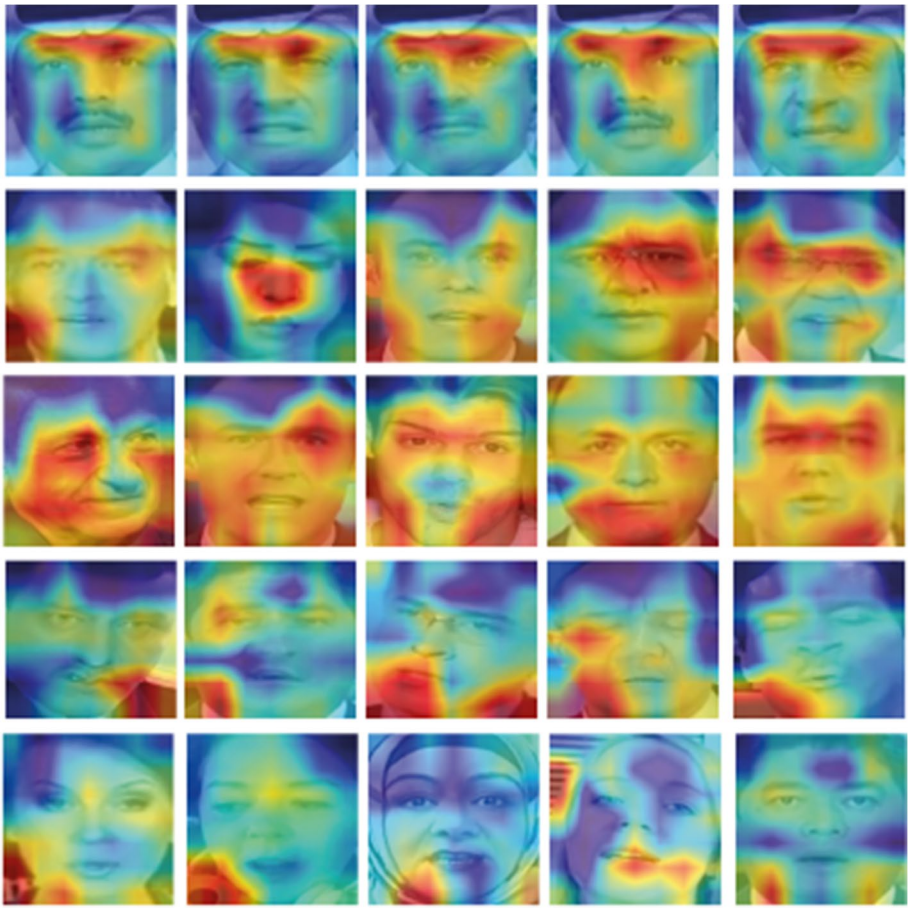

Fig. 9 Each column has different samples of FS, DF, F2F, NT, and SH subsets of FF++datasets. 1st row represents heat maps without post-processing attack, 2nd row represents noise attack heat maps, 3rd row represents blur attack heat maps, 4th row represents pixel dropout attack heat maps, and 5th row represents elastic deformed attack heat maps

the results are given in Table 10. In [20], the FF++dataset was proposed and trained on existing handcrafted and deep learning models like XceptionNet and MesoNet [49]. Of all existing methods, XceptionNet obtained the highest results for each class: FS, DF, F2F, NT, and the combined set. XceptionNet al.so achieved the best results on all the fake classes, except for the real class. For the real class, MesoNet achieved higher accuracy than the other models in [20]. In [51], face swap, deepfake, and face2facesubsets of FF and FF++were used to train a Y-shape decoder for classification and segmentation. The F2F set of the FF++dataset was used for training and the model was tested on the F2F, deepfake, and face swap subsets. When comparing the combined FF++with contemporary methods, it is evident that Steg features [20] achieved the lowest accuracy of 51.8% [60], achieved an accuracy of 97.16%, and our proposed method achieved an accuracy of 96.85%, which is equivalent to the highest accuracy among the listed methods. When compared to state-of-the-art approaches, our method detects all FF++subsets with

Table 10 Comparison of proposed RFE Net with Contemporary methods

Accuracy (%) on FF++, Celeb-DF DFFD, and DFDC Datasets										
Method	Model	FS	DF	F2F	NT	SH	FF++ (Combined)	DFDC	Celeb-DF	DFFD
[20]	XceptionNet Full Image	70.87	74.55	75.91	73.33	-	62.40	-	-	-
	Steg. features	68.93	73.64	73.72	63.33	-	51.80	-	-	-
	Residual based descriptor	73.79	85.45	67.88	78.00	-	55.20	-	-	-
	CNN	56.31	85.45	64.23	60.07	-	58.10	-	-	-
	CNN	82.52	84.55	73.72	70.67	-	61.60	-	-	-
	MesoNet	61.17	87.27	56.20	40.67	-	66.00	-	-	-
	XceptionNet	90.29	96.36	86.86	80.67	-	70.10	-	-	-
[38]	RNA	84.5	94.5	80.3	74.0	-	-	-	66.2	-
[39]	XceptionNet	-	-	-	-	-	-	-	93.7	-
[51]	Classification	54.07	52.30	92.77	-	-	83.71			-
	Segmentation	34.04	70.37	90.27	-	-	93.01			-
[56]	Ensemble of CNNs	-	-	-	-	-	94.0	87.8	-	-
[57]	Capsule Net	99.1	98.6	**99.6**	90.5	95.1	-	-	-	99.6
[58]	DenseNet	95.73	93.9	92.6	83.5	60.90	87.76			-
[59]	Efficient Net	98.89	97.14	98.5	92.11	96.07		88.41		-
[60]	GNN	98.02	98.97	62.49	75.09	97.70	**97.16**	**92.05**	93.90	-
Proposed	**RFE NET**	**99.39**	**98.99**	**98.88**	**92.50**	**97.92**	96.85	90.33	**94.30**	**99.9**

AUC on World Leaders Dataset							
Method	**Subset**	**Obama**	**Clinton**	**Warren**	**Sander**	**Trump**	**Combined**
[31]	FS	0.95	0.95	0.98	0.96	-	0.93
	Imp	0.94	0.93	**1.00**	0.94	0.94	
[43]	-	-	-	**-**	-	-	0.94

Table 10 (continued)

[58]	FS	0.94	0.84	0.93	0.89	-	0.97
	Imp	0.58	0.91	0.93	0.78	0.99	
[60]	FS	1.00	0.90	0.95	**1.00**	-	**0.98**
	Imp	1.00	0.97	0.97	0.91	0.92	
[57]	FS	0.98	0.93	**0.99**	0.99	-	-
	Imp	**1.00**	**0.98**	0.95	0.95	-	
Propose-dRFE NET	FS	**1.00**	**0.98**	0.95	0.95	-	
	Imp	**1.00**	**0.97**	0.96	0.95	**1.00**	0.96

greater precision. Our model captures attribute manipulation artifacts significantly better than other approaches.

When Celeb-DF is compared to contemporary approaches, it is evident that our proposed RFE Net has the highest accuracy among all. However, on the DFDC dataset, the method [60] achieved 2% higher accuracy compared to the proposed method, however use of early stopping regularization makes the proposed method more efficient compared to [60]. This comparison shows that the proposed RFE Net beats existing deepfake detection techniques. These results demonstrate explicitly that the proposed RFE NET can reliably detect numerous types of deepfake.

For the WLDR dataset, the proposed model was compared with the contemporary methods in [31, 43, 57, 58, 60]. The performance of the existing method on this dataset is measured by the AUC, so we compare the AUC of our method with these techniques. Methods [31, 43], performed several experiments to train a single class SVM on subsets of all leaders. The overall average AUC for this model was 0.93. In [43], appearance and behavior bioinformatic features were used for face swap deepfake detection, which achieved an AUC of 94.5 on the WLDR. The method [58] attained less accurate results than the proposed method. In comparison with [31, 43], the proposed game theoretic RFE NET achieved an even higher AUC of 0.99 on the WLDR dataset. Warren's Imposter and faceswap sets [31] and [43], slightly perform better than our method, whereas on the Sander and combined set [60], achieved a performance gain of 2% over the proposed method. Based on this comparison, with the above-mentioned techniques, it can be concluded that the proposed RFE Net outperforms the majority of the contemporary deepfake detection methods and is comparable to [57] and [60]. As clearly shown by these results, the proposed game theoretic RFE NET can effectively be used to detect the diverse types of deepfakes.

5 Conclusion

This paper has presented a novel game theory-based Regularized Forensic Efficient Net model, which combines supervised learning with a game theory approach for video deepfake detection. We presented a zero-sum game for two players: dataset and detector, using different regularization-based strategies to check the generalizability of the player of interest "deepfake detector". The use of mixed strategies allowed the game to achieve NE rewards and our model to better classify diverse types of deepfakes. We performed rigorous experimentation of our method on five benchmark datasets that are FF++, WLDR, CelebDF, DFFD and DFDC against contemporary methods, including cross-set and cross-corpus experiments, and post-processing attacks evaluation to judge the generalizability of the proposed RFE Net. In addition, the explainability factor is also highlighted by extracting the visual artifacts through heat maps. the proposed extensive game allows players to customize the strategies such as regularization making it more generalizable. this adaptability allows the proposed method to detect different deepfakes, improving its effectiveness. Experimental results indicate the effectiveness of the proposed method for diverse types of videos deepfake detection, including identity and expression swapping (face2face, neural textures), and lip-syncing. In the future, we plan to expand our contribution by introducing other game theory concepts with deep learning to improve the generalizability and explainability of deepfake detection.

Acknowledgements This work was supported by the Multimedia Signal Processing (MSP) research lab at the University of Engineering and Technology (UET) Taxila and SMILES lab at Oakland University, USA. We would like to thank Prof. Hany Farid from the University of California Berkeley for providing us with the World Leaders dataset.

Author contributions Ali Javed Conceptualization, Methodology, Validation, Formal analysis, Investigation, Resources, Writing - Original Draft, Writing - Review & Editing, Supervision, Project administration, Funding acquisition. Qurat Ul Ain Methodology, Software, Formal analysis, Investigating, Data Curation, Writing - Original Draft, Writing - Review & Editing, Visualization, Validation. Khalid Mahmood Malik Methodology, Validation, Formal analysis, Investigation, Resources, Writing - Review & Editing. Aun Irtaza Methodology, Investigation, Writing - Review & Editing.

Funding This work was supported by the grant of the Punjab Higher Education Commission (PHEC) of Pakistan via Award No. (PHEC/ARA/PIRCA/20527/21).

Data availability We have provided the links of publicly available datasets used in this work.

Declarations

Competing interests We declared that we have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Dean B (2023) Available: https://backlinko.com/social-media-users. Accessed 20 Jan 2024
2. Antipov G, Baccouche M, Dugelay JL (2017) Face aging with conditional generative adversarial networks. In: 2017 IEEE international conference on image processing (ICIP), IEEE, pp 2089–2093
3. (2023). Available: https://reface.app/guidelines/. Accessed 20 Jan 2024
4. Iperov (2023) Available: https://github.com/iperov/DeepFaceLab. Accessed 20 Jan 2024
5. Masood M, Nawaz M, Malik KM, Javed A, Irtaza A, Malik H (2023) Deepfakes generation and detection: state-of-the-art, open challenges, countermeasures, and way forward. Appl Intell 53(4):3974–4026
6. Kim H, Garrido P, Tewari A, Xu W, Thies J, Niessner M, Pérez P, Richardt C, Zollhöfer M, Theobalt C (2018) Deep video portraits. ACM Trans Graphics (TOG) 37(4):1–14
7. Suwajanakorn S, Seitz SM, Kemelmacher-Shlizerman I (2017) Synthesizing Obama: learning lip sync from audio. ACM Trans Graph (ToG) 36(4):1–13
8. Thies J, Zollhöfer M, Nießner M, Valgaerts L, Stamminger M, Theobalt C (2015) Real-time expression transfer for facial reenactment. ACM Trans Graph 34(6):183–181
9. Thies J, Zollhofer M, Stamminger M, Theobalt C, Nießner M (2016) Face2face: real-time face capture and reenactment of RGB videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2387–2395
10. Nida N, Irtaza A, Ilyas N (2021) Forged face detection using ELA and deep learning techniques. In: 2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST), IEEE, pp 271–275
11. Zhang Y, Zheng L, Thing VL (2017) Automated face swapping and its detection. In 2017 IEEE 2nd international conference on signal and image processing (ICSIP) pp. 15–19. IEEE
12. Javed A, Malik KM (2022) Faceswap deepfakes detection using Novel Multi-directional Hexadecimal Feature Descriptor. In: 2022 19th International Bhurban Conference on Applied Sciences and Technology (IBCAST), IEEE, pp 273–278
13. Farebrother J, Machado MC, Bowling M (2018) Generalization and regularization in dqn. arXiv preprint arXiv:1810.00123
14. Cheng H, Guo Y, Wang T, Nie L, Kankanhalli M (2023) Towards Generalizable Deepfake Detection by Primary Region Regularization. arXiv preprint arXiv:2307.12534
15. Das S, Seferbekov S, Datta A, Islam MS, Amin MR (2021) Towards solving the deepfake problem: An analysis on improving deepfake detection using dynamic face augmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (pp. 3776–3785)
16. Halpern JY (2007) Computer science and game theory: A brief survey. arXiv preprint cs/0703148
17. Von Neumann J, Morgenstern O (1947) Theory of games and economic behavior, 2nd rev

18. Hazra T, Anjaria K (2022) Applications of game theory in deep learning: a survey. Multimed Tools Appl 81(6):8963–8994
19. Janet Chen SI, Vekhter D (2023) 20 July. Available: https://cs.stanford.edu/people/eroberts/courses/soco/projects/1998-99/game-theory/neumann.html
20. Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2019) Faceforensics++: Learning to detect manipulated facial images. In Proceedings of the IEEE/CVF international conference on computer vision (pp. 1–11)
21. Rana MS, Nobi MN, Murali B, Sung AH (2022) Deepfake detection: a systematic literature review. IEEE Access 10:25494–25513
22. Yang X, Li Y, Lyu S (2019) Exposing deep fakes using inconsistent head poses. In ICASSP 2019–2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 8261–8265). IEEE
23. Jack K (2011) Video demystified: a handbook for the digital engineer. Elsevier
24. Ciftci UA, Demir I, Yin L (2020) Fakecatcher: detection of synthetic portrait videos using biological signals. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2020.3009287
25. Jung T, Kim S, Kim K (2020) Deepvision: deepfakes detection using human eye blinking pattern. IEEE Access 8:83144–83154
26. McCloskey S, Albright M (2019) Detecting GAN-generated imagery using saturation cues. In 2019 IEEE international conference on image processing (ICIP) (pp. 4584–4588). IEEE
27. Guarnera L, Giudice O, Battiato S (2020) Deepfake detection by analyzing convolutional traces. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 666–667)
28. Nataraj L, Mohammed TM, Chandrasekaran S, Flenner A, Bappy JH, Roy-Chowdhury AK, Manjunath BS (2019) Detecting GAN generated fake images using co-occurrence matrices. Electron Imaging. https://doi.org/10.2352/ISSN.2470-1173.2019.5.MWSF-532. (arXiv preprint arXiv:1903.06836)
29. Zhang X, Karaman S, Chang SF (2019) Detecting and simulating artifacts in gan fake images. In 2019 IEEE international workshop on information forensics and security (WIFS) pp 1–6. IEEE
30. Amerini I, Galteri L, Caldelli R, Del Bimbo A (2019) Deepfake video detection through optical flow based cnn. In Proceedings of the IEEE/CVF international conference on computer vision workshops (pp 0–0)
31. Agarwal S, Farid H, Gu Y, He M, Nagano K, Li H (2019) Protecting world leaders against deep fakes. In CVPR workshops (Vol. 1, p 38) thecvf.com
32. Baltrušaitis T, Robinson P, Morency LP (2016) Openface: an open source facial behavior analysis toolkit. In: 2016 IEEE winter conference on applications of computer vision (WACV) (pp. 1–10). IEEE
33. Korshunov P, Marcel S (2018) Speaker inconsistency detection in tampered video. In: 2018 26th European signal processing conference (EUSIPCO), IEEE, pp 2375–2379
34. Boutellaa E, Boulkenafet Z, Komulainen J, Hadid A (2016) Audiovisual synchrony assessment for replay attack detection in talking face biometrics. Multimedia Tools Appl 75:5329–5343
35. Li Y, Lyu S (2018) Exposing deepfake videos by detecting face warping artifacts. arXiv preprint arXiv:1811.00656
36. King DE (2009) Dlib-ml: a machine learning toolkit. J Mach Learn Res 10:1755–1758
37. Güera D, Delp EJ (2018) November. Deepfake video detection using recurrent neural networks. In 2018 15th IEEE international conference on advanced video and signal based surveillance (AVSS) (pp 1–6). IEEE
38. Nirkin Y, Wolf L, Keller Y, Hassner T (2021) DeepFake detection based on discrepancies between faces and their context. IEEE Trans Pattern Anal Mach Intell 44(10):6111–6121
39. Chintha A, Thai B, Sohrawardi SJ, Bhatt K, Hickerson A, Wright M, Ptucha R (2020) Recurrent convolutional structures for audio spoof and video deepfake detection. IEEE J Selec Topics Signal Process 14(5):1024–1037
40. Liy CM, InIctuOculi LYUS (2018) Exposingaicreated fakevideosbydetectingeyeblinking. In Proceedings of the 2018 IEEE International workshop on information forensics and security (WIFS), Hong Kong, China (pp 11–13)
41. Montserrat DM, Hao H, Yarlagadda SK, Baireddy S, Shao R, Horváth J, Bartusiak E, Yang J, Guera D, Zhu F, Delp EJ (2020) Deepfakes detection with automatic face weighting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops (pp. 668–669)
42. De Lima O, Franklin S, Basu S, Karwoski B, George A (2020) Deepfake detection using spatiotemporal convolutional networks. arXiv preprint arXiv:2006.14749

43. Agarwal S, Farid H, El-Gaaly T, Lim SN (2020) Detecting deep-fake videos from appearance and behavior. In: 2020 IEEE international workshop on information forensics and security (WIFS), IEEE, pp 1–6
44. Wiles O, Koepke A, Zisserman A (2018) Self-supervised learning of a facial attribute embedding from video. arXiv Preprint arXiv :180806882
45. Rathgeb C, Botaljov A, Stockhardt F, Isadskiy S, Debiasi L, Uhl A, Busch C (2020) PRNU-based detection of facial retouching. IET Biom 9(4):154–164
46. Tariq S, Lee S, Kim H, Shin Y, Woo SS (2018) Detecting both machine and human created fake face images in the wild. In: Proceedings of the 2nd international workshop on multimedia privacy and security (pp. 81–87)
47. Heidari A, Navimipour NJ, Dag H, Talebi S, Unal M (2024) A Novel Blockchain-Based Deepfake Detection Method Using Federated and Deep Learning Models. Cognitive Computation, pp.1–19
48. Marra F, Saltori C, Boato G, Verdoliva L (2019) December. Incremental learning for the detection and classification of GAN-generated images. In 2019 IEEE international workshop on information forensics and security (WIFS) (pp. 1–6). IEEE
49. Afchar D, Nozick V, Yamagishi J, Echizen I (2018) December. Mesonet: a compact facial video forgery detection network. In 2018 IEEE international workshop on information forensics and security (WIFS) (pp. 1–7). IEEE
50. Sabir E, Cheng J, Jaiswal A, AbdAlmageed W, Masi I, Natarajan P (2019) Recurrent convolutional strategies for face manipulation detection in videos. Interfaces (GUI) 3(1):80–87
51. Nguyen HH, Fang F, Yamagishi J, Echizen I (2019) Multi-task learning for detecting and segmenting manipulated facial images and videos. In: 2019 IEEE 10th international conference on biometrics theory, applications and systems (BTAS) (pp 1–8). IEEE
52. Seraj MS, Singh A, Chakraborty S (2024) Semi-Supervised Deep Domain Adaptation for Deepfake Detection. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp 1061–1071
53. Matern F, Riess C, Stamminger M (2019) Exploiting visual artifacts to expose deepfakes and face manipulations. 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). IEEE, pp 83–92
54. Haliassos A, Vougioukas K, Petridis S, Pantic M (2021) Lips don't lie: A generalisable and robust approach to face forgery detection. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5039–5049)
55. Demir I, Çiftçi UA (2024) How Do Deepfakes Move? Motion Magnification for Deepfake Source Detection. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (pp. 4780–4790)
56. Bonettini N, Cannas ED, Mandelli S, Bondi L, Bestagini P, Tubaro S (2021) Video face manipulation detection through ensemble of cnns. In 2020 25th international conference on pattern recognition (ICPR) pp. 5012–5019. IEEE
57. Ilyas H, Javed A, Malik KM, Irtaza A (2023) E-Cap net: an efficient-capsule network for shallow and deepfakes forgery detection. SpringerLink. Multimedia Systems 29(4):2165–2180
58. Khalid F, Javed A, Irtaza A, Malik KM (2023) Deepfakes Catcher: A Novel Fused Truncated DenseNet Model for Deepfakes Detection. In: Proceedings of International Conference on Information Technology and Applications: ICITA 2022 (pp 239–250). Singapore: Springer Nature Singapore
59. Ilyas H, Javed A, Aljasem MM, Alhababi M (2023) Fused Swish-ReLU efficient-net model for deepfakes detection. In: 2023 9th International Conference on Automation, Robotics and Applications (ICARA), IEEE, pp 368–372
60. Khalid F, Javed A, Ilyas H, Irtaza A (2023) DFGNN: An interpretable and generalized graph neural network for deepfakes detection. Expert Syst Appl 222:119843
61. Raza MA, Malik KM, Haq IU (2023) HolisticDFD: infusing spatiotemporal transformer embeddings for deepfake detection. Inf Sci 645:119352. https://doi.org/10.1016/j.ins.2023.119352
62. Zhang D, Zhang H, Zhou H, Bao X, Huo D, Chen R, Cheng X, Wu M, Zhang Q (2021) Building interpretable interaction trees for deep nlp models. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 35, no 16, pp 14328–14337
63. Zhang H, Li S, Ma Y, Li M, Xie Y, Zhang Q (2020) Interpreting and boosting dropout from a game-theoretic view. arXiv preprint arXiv:2009.11729
64. Dong S, Wang J, Liang J, Fan H, Ji R (2022) Explaining deepfake detection by analysing image matching. In: European Conference on Computer Vision, Cham: Springer Nature Switzerland, pp 18–35
65. Wu D, Lisser A (2023) CCGnet: a deep learning approach to predict Nash equilibrium of chance-constrained games. Inf Sci 627:20–33

66. Tembine H (2019) Deep learning meets game theory: Bregman-based algorithms for interactive deep generative adversarial networks. IEEE Trans Cybernetics 50(3):1132–1145
67. Yasodharan S, Loiseau P (2019) Nonzero-sum adversarial hypothesis testing games. In: 2019 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada. Adv Neural Inf Process Syst 32:1–11
68. Sanchez IE (2016) Optimal threshold estimation for binary classifiers using game theory. F1000Research. https://doi.org/10.12688/f1000research.10114.2. (F1000Research)
69. Wu D, Lisser A (2022) Using CNN for solving two-player zero-sum games. Expert Syst Appl 204:117545
70. Couellan N (2017) A note on supervised classification and Nash-equilibrium problems. RAIRO-Operations Res 51(2):329–341
71. Georgiou HV, Mavroforakis ME (2013) A game-theoretic framework for classifier ensembles using weighted majority voting with local accuracy estimates. arXiv Preprint arXiv :13020540
72. Behpour S, Kitani KM, Ziebart BD (2017) ADA: a game-theoretic perspective on data augmentation for object detection. arXiv Preprint arXiv :171007735
73. Xiang J, Zhu G (2017) Joint face detection and facial expression recognition with MTCNN. In: 2017 4th international conference on information science and control engineering (ICISCE), IEEE, pp 424–427
74. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, IEEE, vol 1, pp I-I
75. Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: International conference on machine learning, PMLR, pp 6105–6114
76. Clevert DA, Unterthiner T, Hochreiter S (2015) Fast and accurate deep network learning by exponential linear units (elus). arXiv preprint arXiv:1511.07289
77. Prechelt L (2002) Early stopping-but when? In: Neural networks: tricks of the trade. Springer Berlin Heidelberg, Berlin, Heidelberg, pp 55–69
78. Casado-García Á, Domínguez C, García-Domínguez M, Heras J, Inés A, Mata E, Pascual V (2019) CLoDSA: a tool for augmentation in classification, localization, detection, semantic segmentation and instance segmentation tasks. BMC Bioinform 20:1–14
79. Dolhansky B, Howes R, Pflaum B, Baram N, Ferrer CC (2019) The deepfake detection challenge (dfdc) preview dataset. arXiv preprint arXiv:1910.08854
80. Li Y, Yang X, Sun P, Qi H, Lyu S (2020) Celeb-df: a large-scale challenging dataset for deepfake forensics. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3207–3216. thecvf.com
81. Dang H et al (2020) On the detection of digital face manipulation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern recognition, pp 5781–5790. thecvf.com
82. Ilyas H, Irtaza A, Javed A, Malik KM (2022) Deepfakes examiner: An end-to-end deep learning model for deepfakes videos detection. In: 2022 16th International Conference on Open Source Systems and Technologies (ICOSST), IEEE, pp 1–6
83. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision, pp 618–626. thecvf.com

Authors and Affiliations

Qurat Ul Ain[1] · Ali Javed[2] · Khalid Mahmood Malik[3] · Aun Irtaza[1]

✉ Ali Javed
ali.javed@uettaxila.edu.pk

Qurat Ul Ain
quratul.ain2@students.uettaxila.edu.pk

Khalid Mahmood Malik
drmalik@umich.edu

Aun Irtaza
aun.irtaza@uettaxila.edu.pk

[1] Department of Computer Science, University of Engineering and Technology-Taxila, Taxila 47050, Punjab, Pakistan

[2] Department of Software Engineering, University of Engineering and Technology-Taxila, Taxila 47050, Punjab, Pakistan

[3] College of Innovation and Technology, University of Michigan-Flint, Flint, MI 48502, USA