On Improving Cross-dataset Generalization of Deepfake Detectors

Aakash Varma Nadimpalli and Ajita Rattani
School of Computing
Wichita State University, Wichita, USA
axnadimpalli@shockers.wichita.edu, ajita.rattani@wichita.edu

Abstract

Facial manipulation by deep fake has caused major security risks and raised severe societal concerns. As a countermeasure, a number of deep fake detection methods have been proposed recently. Most of them model deep fake detection as a binary classification problem using a backbone convolutional neural network (CNN) architecture pretrained for the task. These CNN-based methods have demonstrated very high efficacy in deep fake detection with the Area under the Curve (AUC) as high as 0.99*. However, the performance of these methods degrades significantly when evaluated across datasets. In this paper, we formulate deep fake detection as a hybrid combination of supervised and reinforcement learning (RL) to improve its cross-dataset generalization performance. The proposed method chooses the top-*k *augmentations for each test sample by an RL agent in an image-specific manner. The classification scores, obtained using CNN, of all the augmentations of each test image are averaged together for final real or fake classification. Through extensive experimental validation, we demonstrate the superiority of our method over existing published research in cross-dataset generalization of deep fake detectors, thus obtaining state-of-the-art performance.*

1. Introduction

Synthesized media, called deep fakes [7, 23, 31], containing facial information generated by digital manipulation techniques, have become a major political and societal threat [5]. This term "deep fake" signifies deep adversarial models that generate fake content by *swapping* a person's face with the face of another person using deep fake generation techniques such as FaceSwap and FaceShifter [16], [15]. The use of deep fakes to commit fraud, falsify evidence [12], manipulate public debates, and destabilize political processes has raised a top security concern.

To mitigate the risk posed by deep fakes, the popular deep fake detection methods include training convolutional neural networks (CNNs) (such as ResNet-50 [11], XceptionNet [6], and InceptionNet [28]) for detecting facial manipulations from real imagery. These CNNs are trained to detect visual artifacts or blending boundary of a forged image. The deep fake datasets such as Celeb-DF [19], FaceForensics++ [26], DeeperForensics-1.0 [13] and DFDC [9] have been assembled for research and development in detecting deep fakes. These datasets can be categorized into different generations depending on the quality and the amount of the real and fake data.

Most of the aforementioned CNN-based deep fake detection methods obtain very high performance in the intra-dataset evaluation (i.e., when the same dataset is used for training and testing). However, they obtain poor generalization across datasets. For instance, XceptionNet for deep fake detection obtained Area Under Curve (AUC) of 0.997 when trained and tested on FaceForensics++ dataset. However, the AUC score reduced to 0.482 when tested on Celeb-DF in cross-dataset scenario [6]. The degradation in the performance across datasets is due to **domain shift** i.e., the data distribution change between the training and testing set. This is due to change in the image quality of real videos and deep fakes following the continuous advances in sensor technology and the deep fake generation techniques.

The aim of this paper is to *improve the cross-dataset generalization* of the current CNN-based deep fake detectors. To this front, a hybrid combination of the supervised and deep reinforcement learning (RL) module is proposed. Reinforcement learning (RL) [4, 32] is a branch of machine learning that studies how an intelligent agent should operate in a given environment to maximize the concept of cumulative reward. In a standard reinforcement learning model, the agent receives an input, which is the current state of the environment. The agent then chooses an action that changes the current state and the value of the state transition is communicated to the agent through a scalar reinforcement signal called reward. Through systematic trial and error, the agent learns a policy to choose actions that tend to maximize the cumulative reward.

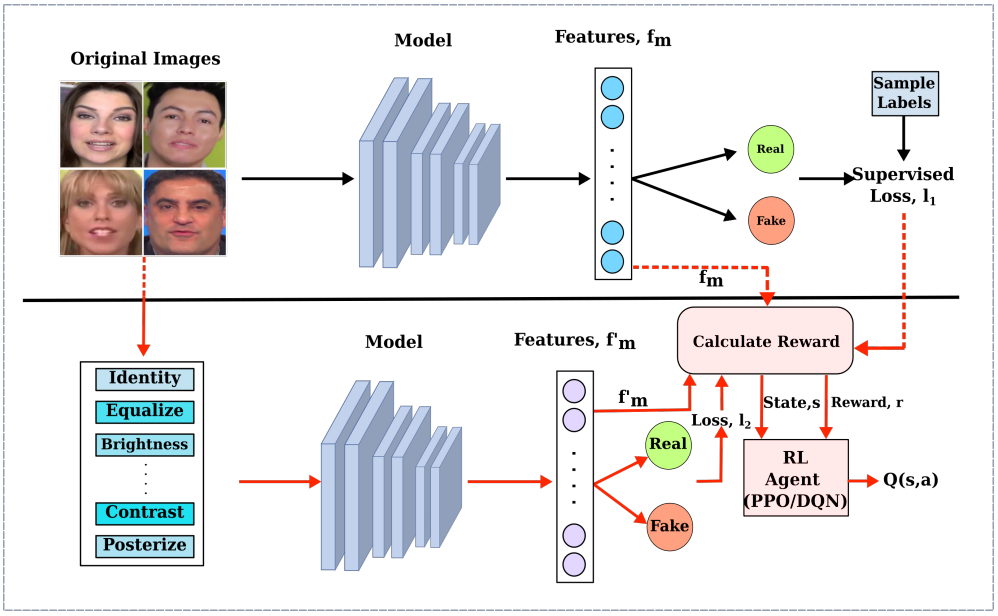


Figure 1. Illustration of the combination of supervised and deep reinforcement learning for training the RL agent for top-k test-time augmentations selection based on the learned policy.

Our proposed approach can meticulously choose and apply top-k augmentations to the test samples via a learned policy by an RL agent in an image-specific manner. The scores of the augmented test samples are averaged together for the final classification. Our experimental results demonstrate the *merit* of test-time image-specific augmentations in *reducing the impact of domain shift* in cross-dataset performance evaluation of deep fake detectors. Experimental investigation on FaceForensics++, Celeb-DF, and DeeperForensics-1.0 show that our approach obtains state-of-the-art performance in cross-dataset generalization of the deep fake detectors over other published work.

In summary, the major **contributions** of this paper are threefold as given below:

- We propose a hybrid combination of supervised and reinforcement learning techniques that applies top-k augmentations to the test samples in an image-specific manner using a policy learned by an RL agent, for deep fake classification.

- We demonstrate the merit of our approach in reducing the impact of domain shift over random test-time data augmentations in performance evaluation of deep fake detectors across datasets.

- Extensive experiments demonstrate the efficacy of our approach in improving the cross-dataset generalization of deep fake detection over published work, thus obtaining state-of-the-art performance in deep fake detection across datasets over existing classification baselines.

This paper is organized as follows: Section 2 discusses the prior work on deep fake detection. Section 3 describes our proposed method for training the RL agent and choosing the top-k augmentations during the test-time for deep fake classification. Datasets and experimental protocol are discussed in section 4. Experimental results are discussed in section 5. The ablation study for choosing the top-k augmentations is detailed in section 6. Conclusion and future work are discussed in section 7.

2. Prior Work on Deepfake Detection

In this section, we will discuss the existing countermeasure proposed for deep fake detection. Most of the existing methods are CNN-based classification baselines trained for deep fake detection [6, 11, 22, 25].

In [18], Li and Lyu used VGG16, ResNet50, ResNet101, and ResNet152 based CNNs for the detection of the presence of artifacts from the facial regions and the surrounding areas for deep fake detection. Afchar et al. [1] proposed two different CNN architectures composed of only a few layers in order to focus on the mesoscopic properties of the images: (a) a CNN comprised of 4 convolutional layers followed by a fully-connected layer (Meso-4), and (b) a modification of Meso-4 using a variant of the Inception module named MesoInception-4. Zhou et al. [35] proposed a two-stream network for face manipulation detection. In particular, the authors considered a fusion of a face classi-

fication stream based on the CNN GoogLeNet and a path triplet stream that is trained using steganalysis features of images patches. In [26], an exhaustive analysis of different CNN-based deep fake detection methods by Rosslet et al. suggested efficacy of XceptionNet when evaluated on FaceForensics++.

Nguyen et al. [22] proposed a multi-task CNN to simultaneously detect the fake videos and locate the manipulated regions using an autoencoder with a Y-shaped decoder for information sharing between classification, segmentation, and reconstruction tasks. In [25], biometric tailored loss functions (such as Center, ArcFace, and A-Softmax) are used for two-class CNN training for deep fake detection. In [17], a face X-ray model has been proposed to detect forgery by detecting the blending boundary of a forged image using a two-class CNN model trained end-to-end.

Apart from the aforementioned CNN-based deep fake detection methods, facial and behavioral biometrics (i.e., facial expression, head, and body movement) have been used for deep fake detection [2, 3, 10, 25]. Study in [20] used a two-branch representation extractor that combines information from the color and the frequency domain using a multi-scale Laplacian of Gaussian (LOG) operator. Very recently in [34], a multi-attentional deep fake detection based on multiple spatial attention heads along with feature aggregation guided by the attention maps is proposed.

Readers are referred to published survey in [31], [23] for detailed information on deep fake detection methods.

3. Proposed Method

In this section, we provide a detailed explanation of the proposed method which works as follows: during the *training stage*, the deep CNN is trained on live and fake images for deep fake classification. Using the hybrid combination of supervised learning (i.e., the CNN output) and reinforcement learning, RL agent is trained for optimum augmentations (action) selection in an image-specific manner.

Figure 1 illustrates the steps involved in training the RL agent for optimum action selection. For each training sample, the CNN outputs the deep feature map (f_m) and the associated loss (l_1) which is the cross-entropy loss. This is followed by the reinforcement learning module. A typical RL technique is as follows: A RL agent receives reward r from the environment along with the current state s. Depending on the inputs (r, s), the RL agent learns a policy to select the right action a for the environment that maximizes the cumulative reward and finally generates the Q-table containing the maximum expected reward for each state, action (s, a) pairs [14] [32].

In the context of our application, the RL module is composed of seven main components namely:

- **Environment**: The CNN model
- **State (s)**: Feature map (f_m) obtained from CNN model
- **Action (a)**: Data augmentations
- **RL agent**: RL agent takes state, reward as input and generates state-action pairs
- **Reward (r)**: Reward plays a key role in adjusting agents policy $\Pi_\Theta(a, s)$
- **Policy** ($\Pi_\Theta(a, s)$): Useful for calculating Q-table
- **Q-table** (Q(s,a)): It contains values for state-action pairs

An action (a) is chosen from a bank of ten augmentations. The training image is then subjected to the action permutation. The permuted image's new feature map f'_m is then extracted from the CNN along with the associated loss, l_2. On comparing loss values l_1,l_2 of the selected image before and after applying the action permutation, the current state (feature map) and the reward (r) is obtained using equation 1. As given in equation 1, if $l_2 < l_1$ the current state will be f'_m and the reward will be l_1-l_2. Otherwise, if $l_2 \geq l_1$ the current state will be f_m and the reward will be calculated as l_1-l_2.

$$reward(r) = \begin{cases} l_1 - l_2, & if \quad l_2 < l_1 \rightarrow f'_m \\ l_2 - l_1, & if \quad l_2 \geq l_1 \rightarrow f_m \end{cases} \tag{1}$$

This process is repeated for each training sample and the policy $\pi_\theta(a, s)$ is learned by the RL agent based on the state (s) of the environment and the calculated reward (r). The final output of the RL agent is a Q-table, $Q(s, a)$, which contain cumulative reward for state-action pairs based on the learned policy $\Pi_\Theta(a, s)$.

During the *testing stage*, the feature map of the test sample obtained by the CNN is given as an input (current state s) to the RL agent. The RL agent outputs the scores from Q-table for each action (augmentation) based on the learned policy. The top-k augmentations are applied to the test sample based on the scores obtained from the Q-table. The classification scores obtained from the trained CNN for each k^{th} augmentation of the test sample are averaged together for the final real/fake classification.

In our experiments, we used 10 augmentations namely, 'Identity', 'AutoContrast', 'Equalize',' Rotate', 'Solarize', 'Color', 'Posterize',' Contrast', 'Brightness', 'Sharpness',' ShearX', 'ShearY', 'TranslateX', 'TranslateY' as the actions (a) to be performed by the RL agent. We evaluated Proximal Policy Optimization (PPO) [27] and Deep Q Network (DQN) [21] based RL agents for learning the policy

and choosing the right actions (a). These RL agents are explained below.

3.1. Proximal Policy Optimization (PPO)

In PPO algorithm [27], there is an actor and a critic model as illustrated in Figure 2. The role of the actor corresponds to the policy π and is used to choose the action for the agent and update the policy network. The critic corresponds to the value function $Q(s,a)$ for an action value or $V(s)$ for a state value. PPO policy would be learned by calculating an estimator of the policy gradient and using it with a stochastic gradient descent algorithm for approximating the function. The most widely used gradient estimator is defined as:

$$\hat{g} = \hat{\mathbb{E}}_t \left[\bigtriangledown_\Theta \log \pi_\Theta \left(a_t|s_t\right) \hat{A}_t \right] \tag{2}$$

where π_θ is a stochastic policy and $\hat{A}_t$ is the estimator of the advantage function at timestep t. Here $\hat{\mathbb{E}}_t$ is the empirical average over a finite batch of samples. The objective function is determined with the help of this gradient estimator and the aim of objective function is to maximize the size of policy update. Finally, the updated policy is used to predict the action (a) which is given as an input to the model or environment. Among all the objective functions, the best surrogate objective function is $L_t^{CLIP}(\theta)$ which obtained highest average normalized score for the PPO algorithm over other surrogate objectives. Therefore, we used $L_t^{CLIP}(\theta)$ for this study given in [27].

3.2. Deep Q Network (DQN)

In Deep Q Network learning (DQN) [21], a neural network is learned to approximate the Q-value function as seen in Figure 3. The state (s) is given as an input and the output is the Q-value of all potential actions (which are the set of augmentations in this study). The optimal action-value function obeys an important identity known as the Bellman equation. A Deep Q-network [21] can be trained by minimizing a sequence of loss functions $L_i(\theta_i)$ that changes at each iteration i and the objective function is defined as:

$$L_i\left(\theta_i\right) = \mathbb{E}_{s,a\sim\rho(.)} \left[\left(y_i - Q\left(s,a;\theta_i\right)\right)^2 \right] \tag{3}$$

where $y_i = \mathbb{E}_{s^{'}\sim\varepsilon} \left[r + \gamma max_{a^{'}} Q(s^{'},a^{'};\theta_{i-1}) | s,a \right]$ is the target for iteration i and $\rho(s)$ is a probability distribution over state s and actions a that is referred to as the behaviour distribution. On differentiating the objective function, the policy gradient π_θ is obtained to learn a policy that uses the optimal strategy to select the optimal action for maximizing the reward. The final Q-table will be a function of $Q^*(s,a)$ which is defined as:

$$Q^*\left(s,a\right) = \mathbb{E}_{s^{'}\sim\varepsilon} \left[r + \gamma max Q^*(s^{'},a^{'}) | s,a \right] \tag{4}$$

The γ is a hyperparameter which is a fixed value and r is the reward that needs to be maximized. In this study, γ is set to 0.5 based on empirical evidence.

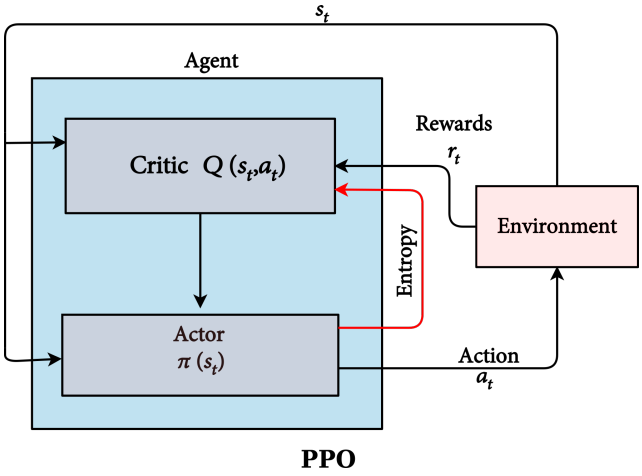


Figure 2. Illustration of the Proximal Policy Optimization (PPO) based RL agent. PPO is based on actor and critic models which helps to generate a policy gradient and a Q-table for the environment. The actor and critic components are helpful to learn the policy $\pi_\theta(s_t)$ and generate the $Q(s_t,a_t)$. An action a_t is selected for the environment by the PPO by time t.

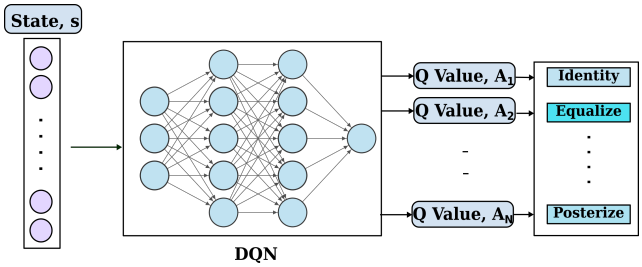


Figure 3. Illustration of the Deep Q Network (DQN) based RL approach which uses a neural network to approximate the Q-value function. The state (s) is given as an input, and the output is the Q-value of all potential actions (a). In this study, state s is the feature map and the set of actions (a) are the set of augmentations (A_i).

4. Dataset and Experimental Protocol

In all the experiments, we used state-of-the-art FaceForensics++ [26], Deeper Forensics-1.0 [13] and Celeb-DF [19] datasets. These datasets are discussed as follows:

- **FaceForensics++:** FaceForensics++ [26] is an automated benchmark for facial manipulation detection. It consists of several manipulated videos created using two different generation techniques: Identity Swapping (FaceSwap, FaceSwap-Kowalski, FaceShifter,

deep fakes) and Expression swapping (Face2Face and NeuralTextures). We used the FaceForensics++ dataset's $c23$ version for both training and testing, which has a curated list of 70 videos for each of these deep fake creation methods.

- **Celeb-DF:** The Celeb-DF [19] deep fake forensic dataset include 590 genuine videos from 59 celebrities as well as 5639 deep fake videos. Celeb-DF, in contrast to other datasets, has essentially no splicing borders, color mismatch, and inconsistencies in face orientation, among other evident deep fake visual artifacts. The deep fake videos in Celeb-DF are created using an encoder-decoder style model which results in better visual quality.

- **DeeperForensics-1.0:** The DeeperForensics-1.0 [13] is one of the largest deep fake dataset used for face forgery detection. DF-1.0 consists of $60,000$ videos that have around 17.6 million frames with substantial real-world perturbations. The dataset contains videos of 100 consented actors with 35 different perturbations. The real to fake videos ratio is 5:1 and the fake videos are generated by an end-to-end face-swapping framework.

Experimental Protocol: We evaluated four different CNN architectures namely, ResNet-50 [11], InceptionNet-v3 [28], EfficientNet v2-L [30] and XceptionNet [6] for deep fake detection. These models were trained on FaceForensics++ $c23$ version which is a high quality (HQ) version of FF++. The face images were detected and aligned using MTCNN [33]. MTCNN utilizes a cascaded CNNs based framework for joint face detection and alignment. The images are then resized to 256×256 for both training and evaluation.

For all the CNN models, we used a batch-normalization layer followed by the last fully connected layer of size 1024 and the final output layer for deep fake classification. The CNN models are trained using an Adam optimizer with an initial learning rate of 0.001 and a weight decay of 1e6. The models are trained on 4 RTX 5000Ti GPUs with a batch size of 64. The same training set is used for training the RL agents for learning the optimum policy using our proposed model.

We used the sampling approach described in [26] to choose 270 frames per video for training and 150 frames per video for validation and testing the models. The trained models are tested on FaceForensics++(HQ), Deeper Forensics-1.0, and Celeb-DF datasets. The standard performance metrics used for deep fake detection namely, Area under the Curve (AUC), Partial Area under the Curve (pAUC) at 10% False Positive Rate (FPR), and the Equal Error Rate (EER) are computed at frame level for the evaluation.

5. Results and Discussion

In this section, we discuss the CNN-based deep fake classification baselines in the intra- and cross-dataset evaluation with and without random test-time augmentations in subsection 5.1. Evaluation results of our proposed hybrid model using PPO and DQN based RL agents in the intra- and cross-dataset scenario are discussed in subsection 5.2. In subsection 5.3, we compare the performance of the existing published results on deep fake detection across datasets with our best results.

5.1. Cross-dataset Generalization of CNN-based Deepfake Detectors

Table 1 shows the performance of the CNN-based deep fake detectors trained on FaceForensics++ and tested on FaceForensics++, DeeperForensics-1.0 and Celeb-DF. These performances are reported with and without test time augmentations. For test-time augmentations, three random augmentations are applied to each test sample and the classification scores are averaged for deep fake detection.

The top performance results are highlighted in bold across various evaluation datasets. Mostly, Efficient v2-L obtained the best results with an AUC of 0.991 and EER of 0.024 when trained and evaluated on FaceForensics++ (intra-dataset). On average, AUC, pAUC, and EER of 0.960, 0.927, and 0.093, respectively, were obtained across the models in intra-dataset evaluation without test-time augmentations.

Across datasets, the performance of all the *models* dropped significantly. On average, AUC, pAUC, and EER of 0.905, 0.844, and 0.175 were obtained across the models when trained on FaceForensics++ and evaluated on DeeperForensics-1.0. On the Celeb-DF dataset, on average, AUC, pAUC, and EER of 0.633, 0.60, and 0.406, respectively, were obtained across the models when trained on FaceForensics++ and evaluated on Celeb-DF.

Mostly, the EfficientNet V2-L backbone obtained the best performance in the intra- and cross-dataset. The *performance drop of the models across datasets is significant for Celeb-DF over DeeperForensics*-1.0. The reason being deep fake videos in Celeb-DF are created using an encoder-decoder style model which results in better visual quality. Whereas, the videos created in DeeperForensics-1.0 uses deep fake generation techniques similar to FaceForenics++.

For most of the models, random test-time augmentations reduced their performance in intra- and cross-dataset evaluation. *This suggests the need for systematic selection of test-time augmentations in an image-specific manner for an enhanced performance*.

5.2. Evaluation of Our Proposed Model

Table 2 shows the deep fake detection performance of our proposed model for both PPO and DQN based RL

Table 1. Evaluation of the existing CNN-based deep fake detection baselines trained on FaceForensics++ and tested on FaceForensics++, DeeperForensics-1.0, and Celeb-DF. In order to establish the merit of our approach of systematic selection of augmentations for test samples by an RL agent, we also evaluated the current deep fake detectors with random test-time augmentations. The top performance results are highlighted in bold.

Models	Test time Augmentation	Training Dataset	Evaluation Datasets								
			FaceForensics++			DeeperForensics-1.0			Celeb-DF		
			AUC	pAUC	EER	AUC	pAUC	EER	AUC	pAUC	EER
ResNet-50	-	FF++	0.945	0.908	0.125	0.885	0.843	0.208	0.625	0.584	0.405
XceptionNet	-	FF++	0.985	0.969	0.037	**0.940**	**0.825**	**0.127**	0.651	0.629	0.383
EfficientNet V2-L	-	FF++	**0.991**	**0.979**	**0.024**	0.938	0.882	0.142	**0.658**	**0.635**	**0.379**
InceptionNet	-	FF++	0.922	0.852	0.187	0.859	0.826	0.225	0.598	0.552	0.459
ResNet-50	✓	FF++	0.919	0.882	0.193	0.849	0.826	0.248	0.592	0.564	0.479
XceptionNet	✓	FF++	0.959	0.927	0.124	**0.912**	**0.875**	**0.197**	0.615	0.589	0.392
EfficientNet V2-L	✓	FF++	**0.967**	**0.946**	**0.118**	0.907	0.864	0.187	**0.622**	**0.598**	**0.385**
InceptionNet	✓	FF++	0.896	0.874	0.195	0.828	0.814	0.275	0.569	0.545	0.495

Table 2. Evaluation of our proposed model in intra and cross-dataset scenarios with PPO and DQN as RL Agents. The top-3 augmentations are selection for each test sample. The classification scores from top-3 augmentations of a test sample are averaged for final deep fake detection. The top performances are highlighted in bold.

Models	RL Methods		Training Dataset	Evaluation Datasets								
	PPO	DQN		FaceForensics++			DeeperForensics-1.0			Celeb-DF		
				AUC	pAUC	EER	AUC	pAUC	EER	AUC	pAUC	EER
ResNet-50	✓	-	FF++	0.948	0.914	0.165	0.897	0.854	0.203	0.629	0.592	0.398
Xception Net	✓	-	FF++	0.989	0.972	0.035	0.944	0.912	0.125	0.657	0.625	0.376
Efficient Net V2-L	✓	-	FF++	**0.994**	**0.983**	**0.022**	**0.952**	**0.922**	**0.119**	**0.669**	**0.647**	**0.362**
Inception Net	✓	-	FF++	0.935	0.886	0.148	0.869	0.847	0.218	0.608	0.576	0.424
ResNet-50	-	✓	FF++	0.937	0.912	0.139	0.879	0.834	0.253	0.619	0.572	0.419
Xception Net	-	✓	FF++	0.979	0.952	0.039	0.937	0.898	0.129	**0.656**	**0.642**	**0.356**
Efficient Net V2-L	-	✓	FF++	**0.982**	**0.965**	**0.039**	**0.948**	**0.917**	**0.122**	0.647	0.628	0.385
Inception Net	-	✓	FF++	0.927	0.899	0.184	0.854	0.822	0.229	0.592	0.549	0.495

agents. The top performance results are highlighted in bold across various datasets. All the results are reported for top-3 augmentations applied to the test samples selected by an RL agent before deep fake classification. The top-3 augmentations are selected based on empirical evidence.

As can be seen from the table, EfficientNet V2-L along with PPO-based RL agent is consistently the best performing model when evaluated across different datasets. This model with PPO obtained an AUC of 0.994 and an EER of 0.022 when trained and evaluated on FaceForensics++.

On average, AUC, pAUC, and EER of 0.966, 0.938, and 0.092 were obtained across the models with PPO-based RL agent in the intra-dataset evaluation. *This suggests that intra-dataset performance of the CNN-based deep fake detectors remain intact when top-k augmentations are applied to test samples by an RL agent using our proposed model.*

Across datasets, EfficientNet v2-L with PPO-based RL agent has obtained the highest AUC and the least EER of 0.952 and 0.119, respectively, when evaluated on DeeperForensics-1.0. When evaluated on Celeb-DF, EfficientNet v2-L with PPO-based RL agent has obtained the highest AUC and the least EER of 0.669 and 0.362, respectively. On average, AUC, pAUC, and EER of 0.640, 0.610, and 0.390 were obtained across the models with the PPO-based RL agent when evaluated on Celeb-DF.

The PPO-based RL agent outperformed DQN for all the models. The reason is as an on-policy algorithm, PPO tackles the problem of sample efficiency by employing surrogate objectives to keep the new policy from drifting too much from the old policy [27]. This aids in learning the optimal policy.

Comparison with the baseline CNNs in Table 1: In comparison to the baseline in Table 1, all the same CNN architectures along with an RL agent for optimum test-time augmentations selection, obtained performance improvement across datasets. On average, an increase in AUC of 0.021, pAUC of 0.031, and a decrease in EER of 0.027 was obtained across all the models using our proposed method on the DeeperForensics-1.0 dataset. Similarly, when evaluated

Table 3. Comparison with existing studies on cross dataset evaluation of deep fake detection on Celeb-DF. The results from existing methods are directly taken from [20]. AUC scores are used for comparison.

Method	FF++	Celeb-DF
Mes04 [1]	0.847	0.548
FWA [18]	0.801	0.569
Xception-raw [19]	0.997	0.482
Xception-c23 [19]	0.997	0.653
Xception-c40 [19]	0.955	0.655
MesoInception4 [1]	0.830	0.536
Two-stream [35]	0.701	0.538
Multi-task [22]	0.763	0.543
Capsule [19]	0.966	0.575
DSP-FWA [18]	0.930	0.646
Two Branch [20]	0.931	0.734
F^3-Net [24]	0.981	0.651
EfficientNet-B4 [29]	0.997	0.642
Ours (EfficientNet-v2-L with PPO)	**0.994**	**0.669**

on Celeb-DF, our proposed model obtained an increase in AUC of 0.018, pAUC of 0.016, and a decrease in EER of 0.021 over the baseline CNNs in Table 1. *Thus, demonstrating the merit of our proposed approach in (a) reducing the impact of domain shift, and (b) transferability of the learned policy towards cross-dataset generalization improvement over existing CNN-based baselines.*

5.3. Comparison with Published Results on Deepfake Detection Across Dataset

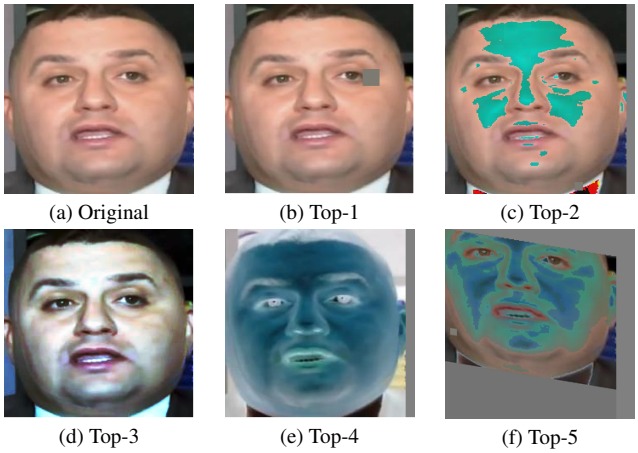
In Table 3, we compared the existing results published in [20] for deepfake detectors trained on FaceForensics++ (FF++) high quality version and tested on FaceForensics++(HQ) and Celeb-DF, with our best results obtained using EfficientNet V2-L with PPO based RL agent (see Table 2). Our proposed best model obtained *equivalent performance* with most of the best-performing methods in intra-dataset evaluation.

At the same time, our best model *outperformed most of the existing methods in cross-dataset evaluation* on CelebDF with an AUC of 0.669. Thus, obtaining state-of-the-art performance. Recall that due to the difference in the deepfake generation techniques between FaceForensics++(HQ) and Celeb-DF dataset, the performance drop of all the models in Table 3 is significant on cross-dataset evaluation.

Although, Two-branch [20] obtained better results with an AUC of 0.734 over our model in cross-dataset evaluation. This method combines representation from color and frequency domain using multi-scale Laplacian of Gaussian (LOG) operator for deep fake detection. Hence, it is not directly comparable to deep learning-based models. Further, its intra-dataset performance is quite low with an AUC of 0.931 over our model with an AUC of 0.994 and other existing methods in Table 3.

Table 4. Ablation study for choosing the optimum top-k augmentations for test samples.

Augmentation Selection	FF++	Celeb-DF
Top-1	0.986	0.665
Top-2	0.988	0.662
Top-3	**0.994**	**0.669**
Top-4	0.967	0.649
Top-5	0.929	0.605



(a) Original (b) Top-1 (c) Top-2

(d) Top-3 (e) Top-4 (f) Top-5

Figure 4. The top-k augmentations with k ranging from 1 to 5 selected by PPO-based RL agent for a fake image from the FaceForensics++ dataset. These augmentations are selected from the scores obtained using Q-table generated by the RL agent. The top-3 augmentations obtained better results for most of the models.

6. Ablation Study: Top-k Augmentations Selection

In this section, we did an ablation study to select the optimum k augmentations from the list of augmentations based on the $Q(s, a)$ table that is generated by the RL agent. For the purpose of this experiment, we used the best performing EfficientNet V2-L with PPO as an RL agent trained on Faceforensics++.

Table 4 shows the performance of EfficientNet along with PPO for top 1 to 5 augmentations. It can be seen that optimum results of 0.994 AUC and 0.022 EER are obtained for top-3 augmentations on FaceForensics++. Similarly, the highest AUC of 0.669 and least EER of 0.362 is obtained for top-3 augmentations when tested on Celeb-DF. A similar trend was observed for the DeeperForensics-1.0

testbed. Figure 4 shows the top-5 augmentations selected by PPO-based RL agent for a sample fake image from the FaceForensics++ dataset. In summary, top-3 augmentations obtained the optimum results for deep fake detection using our model.

7. Conclusion and Future Work

In this paper, we improve the cross-dataset generalization of the CNN-based deep fake detectors. The proposed model applies top-k augmentations to the test samples in an image-specific manner using a policy learned by an RL agent to reduce the impact of domain shift across datasets. Experimental results demonstrate the merit of systematic selection of augmentations based on the quality of each test image in cross dataset performance improvement over random augmentations. The proposed model could be used with any deep fake detection method based on supervised learning. As a part of future work, experimental evaluations will be extended with a larger number of augmentations on different datasets. The proposed RL-based method will be evaluated along with deep fake detection based on fine-grained classification [34] and biometric features [3, 10, 25], for cross-dataset performance improvement. Further, cross-comparison of our task-specific RL-based test-time augmentation approach will be conducted with automatic general-purpose data augmentation pipelines [8].

8. Acknowledgement

The research infrastructure used in this study is supported in part from a grant no. 13106715 from the Defense University Research Instrumentation Program (DURIP) from Air Force Office of Scientific Research.

References

[1] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen. Mesonet: a compact facial video forgery detection network. In *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–7, 2018. 2, 7

[2] Shruti Agarwal, Hany Farid, Tarek El-Gaaly, and Ser-Nam Lim. Detecting deep-fake videos from appearance and behavior. In *2020 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2020. 3

[3] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting World Leaders Against Deep Fakes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, page 8, Long Beach, CA, June 2019. IEEE. 3, 8

[4] Kai Arulkumaran, Marc P. Deisenroth, Miles Brundage, and Anil A. Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017. 1

[5] Rory Cellan-Jones. Deepfake videos 'double in nine months', Oct 2019. 1

[6] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, Los Alamitos, CA, USA, jul 2017. IEEE Computer Society. 1, 2, 5

[7] Danielle Citron. How deepfakes undermine truth and threaten democracy. 1

[8] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. In *IEEE Computer Vision and Pattern Recognition*, 2019. 8

[9] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset, 2020. 1

[10] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Dong Chen, Fang Wen, and Baining Guo. Identity-driven deepfake detection. *ArXiv*, abs/2012.03930, 2020. 3, 8

[11] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 1, 2, 5

[12] Tim Hwang. Deepfakes: A grounded threat assessment. Technical report, Georgetown University, July 2020. 1

[13] Liming Jiang, Ren Li, Wayne Wu, Chen Qian, and Chen Change Loy. DeeperForensics-1.0: A large-scale dataset for real-world face forgery detection. In *CVPR*, 2020. 1, 4, 5

[14] Leslie Pack Kaelbling, Michael L. Littman, and Andrew W. Moore. Reinforcement learning: A survey. *J. Artif. Int. Res.*, 4(1):237–285, may 1996. 3

[15] Iryna Korshunova, Wenzhe Shi, Joni Dambre, and Lucas Theis. Fast face-swap using convolutional neural networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3697–3705, 2017. 1

[16] Lingzhi Li, Jianmin Bao, Hao Yang, Dong Chen, and Fang Wen. Faceshifter: Towards high fidelity and occlusion aware face swapping. *ArXiv*, abs/1912.13457, 2019. 1

[17] Lingzhi Li, Jianmin Bao, Ting Zhang, Hao Yang, Dong Chen, Fang Wen, and Baining Guo. Face x-ray for more general face forgery detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5000–5009, 2020. 3

[18] Yuezun Li and Siwei Lyu. Exposing deepfake videos by detecting face warping artifacts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2, 7

[19] Yuezun Li, Xin Yang, Pu Sun, Honggang Qi, and Siwei Lyu. Celeb-df: A large-scale challenging dataset for deepfake forensics. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3204–3213, 2020. 1, 4, 5, 7

[20] Iacopo Masi, Aditya Killekar, Royston Marian Mascarenhas, Shenoy Pratik Gurudatt, and Wael AbdAlmageed. Two-branch recurrent network for isolating deepfakes in videos.

In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII*, page 667–684, Berlin, Heidelberg, 2020. Springer-Verlag. 3, 7

[21] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. Playing atari with deep reinforcement learning. *ArXiv*, abs/1312.5602, 2013. 3, 4

[22] Huy Hoang Nguyen, Fuming Fang, Junichi Yamagishi, and Isao Echizen. Multi-task learning for detecting and segmenting manipulated facial images and videos. *2019 IEEE 10th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pages 1–8, 2019. 2, 3, 7

[23] Thanh Thi Nguyen, Cuong Manh Nguyen, Dung Nguyen, Duc Thanh Nguyen, and Saeid Nahavandi. Deep learning for deepfakes creation and detection. *ArXiv*, abs/1909.11573, 2019. 1, 3

[24] Yuyang Qian, Guojun Yin, Lu Sheng, Zixuan Chen, and Jing Shao. Thinking in frequency: Face forgery detection by mining frequency-aware clues. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII*, page 86–103, Berlin, Heidelberg, 2020. Springer-Verlag. 7

[25] Sreeraj Ramachandran, Aakash Varma Nadimpalli, and Ajita Rattani. An experimental evaluation on deepfake detection using deep face recognition. In *2021 IEEE International Carnahan Conference on Security Technology (ICCST)*, pages 1–6, 2021. 2, 3, 8

[26] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. Faceforensics++: Learning to detect manipulated facial images. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1–11, 2019. 1, 3, 4, 5

[27] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017. 3, 4, 6

[28] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2826, 2016. 1, 5

[29] Mingxing Tan and Quoc Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, 09–15 Jun 2019. 7

[30] Mingxing Tan and Quoc Le. Efficientnetv2: Smaller models and faster training. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10096–10106. PMLR, 18–24 Jul 2021. 5

[31] Rubén Tolosana, Rubén Vera-Rodríguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. Deepfakes and beyond: A survey of face manipulation and fake detection. *Inf. Fusion*, 64:131–148, 2020. 1, 3

[32] Martijn van Otterlo and Marco Wiering. *Reinforcement Learning and Markov Decision Processes*, pages 3–42. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. 1, 3

[33] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10):1499–1503, 2016. 5

[34] Hanqing Zhao, Tianyi Wei, Wenbo Zhou, Weiming Zhang, Dongdong Chen, and Nenghai Yu. Multi-attentional deepfake detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2185–2194, 2021. 3, 8

[35] Peng Zhou, Xintong Han, Vlad I. Morariu, and Larry S. Davis. Two-stream neural networks for tampered face detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1831–1839, 2017. 2, 7