MixNet for Generalized Face Presentation Attack Detection

Nilay Sanghvi[1], Sushant Kumar Singh[1], Akshay Agarwal[1,2], Mayank Vatsa[3], and Richa Singh[3]

[1]IIIT-Delhi, India; [2]Texas A&M University, Kingsville, USA; [3]IIT Jodhpur, India

[1]{nilay16063, sushant16103, akshaya}@iiitd.ac.in; [3]{mvatsa, richa}@iitj.ac.in

Abstract—The non-intrusive nature and high accuracy of face recognition algorithms have led to their successful deployment across multiple applications ranging from border access to mobile unlocking and digital payments. However, their vulnerability against sophisticated and cost-effective presentation attack mediums raises essential questions regarding its reliability. In the literature, several presentation attack detection algorithms are presented; however, they are still far behind from reality. The major problem with existing work is the generalizability against multiple attacks both in the seen and unseen setting. The algorithms which are useful for one kind of attack (such as print) perform unsatisfactorily for another type of attack (such as silicone masks). In this research, we have proposed a deep learning-based network termed as *MixNet* to detect presentation attacks in cross-database and unseen attack settings. The proposed algorithm utilizes state-of-the-art convolutional neural network architectures and learns the feature mapping for each attack category. Experiments are performed using multiple challenging face presentation attack databases such as SMAD and Spoof In the Wild (SiW-M) databases. Extensive experiments and comparison with existing state of the art algorithms show the effectiveness of the proposed algorithm.

I. INTRODUCTION

Face being a non-intrusive biometrics modality has been deployed to various security-related areas ranging from constrained scenarios such as mobile unlocking to unconstrained scenarios such as surveillance. A forecast[1] shows the popularity of face recognition, which claims that the face recognition market will increase to USD 10.9 billion by 2025 as compared to USD 4.4 billion in 2019. However, the significant challenge of the technology is the vulnerability against presentation attacks. For instance, an attacker can hide the identity by merely wearing a mask [1], or an intruder can illegally access the system using a 2D printed photo [2]. The tremendous amount of face images on social media platforms and unrestricted access to them can make it accessible to perform the attack.

The prevalent presentation attacks on face recognition can be broadly classified into 2D artifacts based and 3D artifacts based attacks. 2D attacks cover printed photos using the printer and replay of photos or videos on an electronic screen. 3D attacks such as silicone masks and latex masks are sophisticated attacks and exhibit properties similar to the natural face. In the literature, several presentation attack detection (PAD) algorithms are presented, which are found effective in handling similar domain attacks, i.e., where the detector

Fig. 1. Images of genuine and different types of attack classes. The classification scores computed using DenseNet121 [3] and the proposed algorithm are also written. For genuine class the score should be close to 0 and for attack it should be close to 1 for correct classification.

has seen the attack type or database at the time of training. However, the generalizability against multiple attacks is still a challenging task. At the same time, the development of new sophisticated silicone mask based attacks increases the detection complexity. The effectiveness of these 3D silicone masks can be seen in the following two cases: (i) a young person fooled the airport authority by wearing a mask and boarded the airplane[2] and (ii) face recognition algorithm in iPhone X is fooled by cost-effective masks[3]. On the other hand, 2D based attacking mediums can also be used for illegal access in unattended recognition systems. Therefore, the generalizability of the PAD algorithms across attack types is crucial. The aim of an effective PAD algorithm is to classify the images as genuine or attack in the first step so that the fake data is not processed through the recognition system.

The prime objective of this research is to develop a generalized PAD algorithm. For an effective PAD algorithm, a challenging and unconstrained database is the first necessity. While several databases are presented in the literature, the significant limitation is the amount of data against each attack category. To address this problem, we have merged two challenging databases, namely SMAD [4] and SiW-M [5]. The SMAD database contains silicone mask attack and authentic images captured in unconstrained settings. The SiW-M database is

---

captured in the wild containing multiple attack types, including full masks and half masks. Further, we have also utilized two popular 2D attack databases, namely Replay-Attack [6] and MSU-MFSD [7] for experimentation and comparison with existing works. The experiments are performed in challenging conditions, including seen attacks, unseen attacks, and cross-database settings. As shown in Fig. 1, the proposed algorithm is not only able to classify different images correctly, but also able to handle variations such as pose and illumination.

In brief, the key highlights of this research are:

- A novel face PAD algorithm termed as MixNet is proposed. It consists of three sub-architectures, one each for detecting the three broad face presentation attacks - print, replay, and mask attack;
- The proposed algorithm, unlike existing algorithms, can further identify the type of attack images, i.e., whether the images come from print, replay or mask attack without an extra computational overhead;
- Extensive experiments concerning seen and unseen domains showcase the strength of the proposed algorithm as compared to hand-crafted features based and convolutional neural network (CNN) based PAD algorithms.

II. RELATED WORK

The popular face PAD algorithms can be broadly grouped into pre-deep learning era and post-deep learning era. The pre-deep learning era based algorithms are mainly based on the extraction of texture features [8]–[13], motion cue based [14], [15], and hybrid algorithms [2], [16], [17]. The hand-crafted features based algorithms are computationally efficient and effective for the same domain attacks but lack generalizability against unseen attacks, databases, or even sensors. Moreover, generalizability is not the only issue, as shown by Agarwal et al. [18], [19], existing PAD algorithms can be fooled through feature manipulation or image transformations.

Menotti et al. [20] and Tu and Fang [21] proposed the deep architecture either through optimization or transfer learning to utilize them for face anti-spoofing. Liu et al. [5] have proposed deep tree learning for zero-shot attack detection. Recently, Mehta et al. [22] developed the panoptic face PAD algorithm using a shallow CNN model trained using focal loss. Their algorithm yields high detection accuracy on the individual attack and combined attack databases but lacks generalizability against unseen attack or database. Jia et al. [23] have studied popular features implemented so far for PAD in mobile scenarios using multiple challenging databases. It is found that ResNet50 based detector yields the best result under cross-database testing. Furthermore, several survey papers have discussed existing PAD algorithms along with their limitations [1], [24].

In the literature, it is observed that most of the existing algorithms are useful for a particular kind of attack or database, but are less effective against multiple attacks in generalized settings. Therefore, in this research, with the aim of generalizability across multiple attacks, both in seen and unseen settings, a novel CNN architecture based PAD algorithm is presented. For each broad category of attack, a CNN architecture is deployed for feature learning and at the end confidence scores are combined together to yield the final detection result.

III. PROPOSED FACE PAD ALGORITHM: MIXNET

As explained in Section I, face presentation attacks can be broadly classified into three categories: print, replay, and 3D mask attacks. However, recent databases also contain the variations of these attacks such as half masks, paper masks, and transparent masks. An effective face PAD algorithm should be agnostic to these variations while detecting the traditional presentation attacks.

Most of the current algorithms have posed face PAD as a binary classification problem, and the algorithms learn to differentiate only between genuine and not genuine (i.e., attack) samples. It might be the reason because of which most of the existing algorithms are not generalized against unseen attacks. The characteristics of the print attack (hard surface, glossy, 2D) are entirely different from that of mask attack (smooth texture, 3D, similar to the skin); therefore, learning a single unified network is challenging. In the proposed algorithm, termed as MixNet, we have added an intermediate step of detecting the three broad attacks before the final classification of genuine/attack. MixNet consists of three sub-architectures where each of them learns the feature mapping of one of the three broad attacks.

A. Training of MixNet

On passing an attack sample to MixNet, only the sub-architecture responsible for detecting that attack should output a score close to 1. In contrast, the other two sub-architectures should output a score close to 0. Finally, after combining these three scores, MixNet shall return the final classification score close to 1 to denote the detection of an attack. To enforce the above process while training the architecture, we use four losses and label each data sample as a quadruple, which we explain in the following subsections III-A1, III-A2.

1) Loss Function: Each sub-architecture has a loss associated with it, which results in three losses - *'print loss'*, *'replay loss'*, and *'mask loss'*. A particular loss enforces the corresponding sub-architecture to detect the associated attack with high efficiency. Further, there is **final classification loss** for the output layer (softmax) of MixNet to classify a sample as genuine or an attack. During training, MixNet tries to minimize the total loss:

$$\mathcal{L}_{total} = \alpha_1 \mathcal{L}_{print} + \alpha_2 \mathcal{L}_{replay} + \alpha_3 \mathcal{L}_{mask} + \alpha_4 \mathcal{L}_{final} \quad (1)$$

where $\alpha_1, \alpha_2, \alpha_3, \alpha_4$ are the regularization coefficients for the four losses. Each of these losses is categorical cross-entropy loss represented as:

$$\mathcal{L}_{cross-entropy} = -\sum_i y_i \log(p_i) \quad (2)$$

Fig. 2(a) shows the forward and the backward pass of the proposed MixNet. When an image is forward passed, it goes
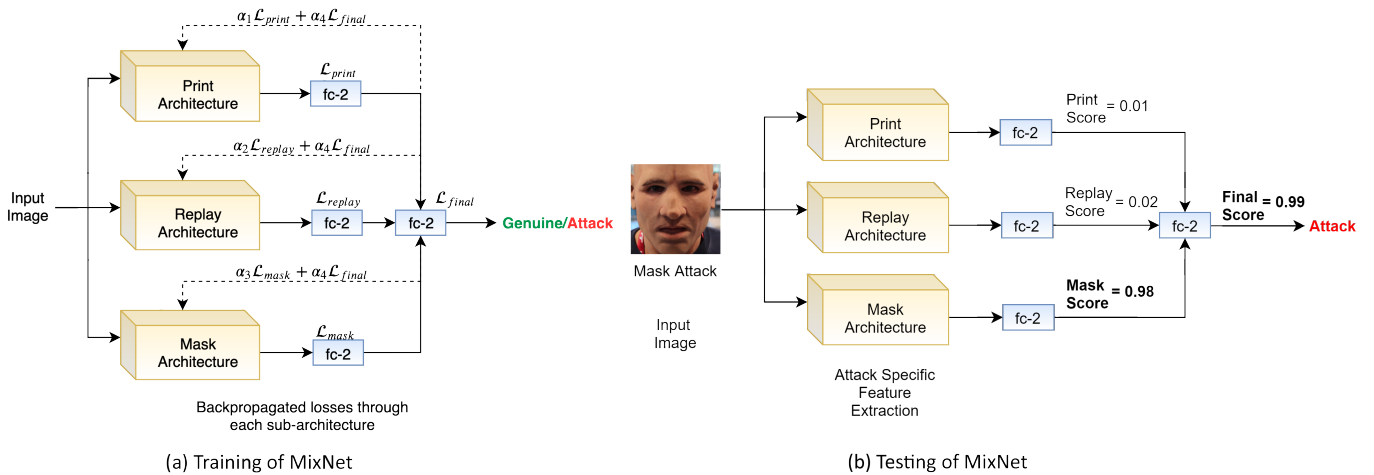
Fig. 2. Schematic diagram of the proposed MixNet for face presentation attack detection.

through each architecture and based on the label of the input image, the amount of loss corresponding to each branch is back-propagated. Each loss affects only the layers that connect the input to the loss. For example, during backpropagation *'print architecture'* would only be affected by *'print loss'* and *'final classification loss'*, i.e.,

$$\alpha_1 \mathcal{L}_{print} + \alpha_4 \mathcal{L}_{final} \qquad (3)$$

2) Architecture Details: For training the proposed MixNet, we label each data sample as described in Table I. The first three entries correspond to the desired output from the three sub-architectures (print, replay, and mask architectures). On the other hand, the last entry corresponds to the final classification output of MixNet.

The recent PAD algorithms [23], [25], [26] yields state-of-the-art performance when deep CNNs such as ResNet50 are used as the base network. Inspired from these studies, in this research, we have used three different deep CNN models for the three sub-architectures: ResNet50 [27] (pre-trained on ImageNet [28]), ResNet50-VF2 (pre-trained on VG-GFace2 [29]) and DenseNet121 [3] (pre-trained on ImageNet). The MixNet with these architectures are referred MixNet-ResNet50, MixNet-ResNet50-VF2 and MixNet-DenseNet121, respectively. The regularization coefficients for each network are as follows: (i) MixNet-ResNet50: $\alpha_1 = 0.3$, $\alpha_2 = 0.5$, $\alpha_3 = 1.0$, $\alpha_4 = 5.0$; (ii) MixNet-DenseNet121 and MixNet-ResNet50-VF2: $\alpha_1 = 0.33$, $\alpha_2 = 0.33$, $\alpha_3 = 0.33$, $\alpha_4 = 5.0$. The coefficients were experimentally found using grid-search

$(\alpha_{1..3} = [0,1]$ and $\alpha_4 = [0,10])$ on the training set. The networks are trained with a batch size of 16 to optimize the loss mentioned in section III-A1 using SGD optimizer with the learning rate of 0.01.

B. Testing of MixNet

Once the MixNet is trained, it is utilized for detecting different attacks and genuine samples. *'print architecture'* would learn to detect print attacks. Similarly, *'replay architecture'* and *'mask architecture'* would learn to detect replay and mask attacks, respectively. Each sub-architecture outputs a score between 0 and 1, which indicates the confidence that the corresponding attack is present in the input image. For the final classification, MixNet combines the scores from the three sub-architectures. As shown in Fig. 2(b), when the input is mask attack image, mask architecture yields a score close to 1 while print and replay architectures output a score close to 0. In the end, the final softmax layer yields the score close to 1, which implies the input is an attack image.

IV. EXPERIMENTAL SETTINGS

In this section, we describe the details of the experimental evaluations. Three popular presentation attacks, i.e., print, replay, and 3D mask, are utilized in our experiments. Print and replay attacks are cost-effective and simple to perform but are not as effective as 3D mask attacks, especially silicone masks, which are significantly costlier and developed using sophisticated hardware and software. For the efficacy of the proposed algorithm we have used two challenging databases; namely, Silicone Mask Attack Database (SMAD) and Spoof In the Wild with Multiple Attack Types (SiW-M). We have merged these databases together to create a large scale database effective for training. Next, each database is described followed by the description of experimental protocols and evaluation metrics used to report the results.

TABLE I
LABELING OF THE INPUT DATA FOR TRAINING MIXNET.
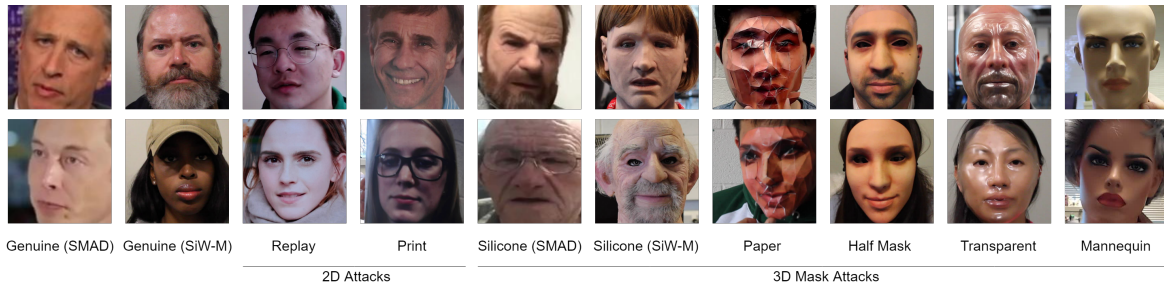
Type of Sample	Print Label	Replay Label	Mask Label	Final Label
Genuine	0	0	0	0
Print Attack	1	0	0	1
Replay Attack	0	1	0	1
Mask Attack	0	0	1	1

Fig. 3. The examples of genuine and attack images from the merged database.

Video Type	Database	Number of Videos
Genuine	SMAD	65
Genuine (from train split)	SiW-M	217
Print Attack	SiW-M	104
Replay Attack	SiW-M	99
Mask Attack	SMAD	65

Video Type	Number of Videos	Scenario
Genuine (from test split)	131	Seen
Silicone Mask	27	Cross
Paper Mask	17	Unseen
Half Mask	72	Unseen
Transparent Mask	88	Unseen
Mannequin	40	Unseen

A. Databases

SMAD: Manjani et al. [4] created this first real-life silicone mask attack database consisting of a total of 130 genuine and mask attack videos. Amongst the 65 real and 65 attack videos, 43 and 59 are males, respectively; rest belong to females. The authors have collected these videos from multiple sources on the web. Thus, it has many variations in background, illumination, facial expression, and video quality, making it a challenging database.

SiW-M: Liu et al. [5] introduced SiW-M, which contains a total of $1,630$ videos of 5-7 seconds each. It consists of 968 videos of 13 different attack types and 660 authentic videos from 493 subjects. The attack types include print attack, replay attack, and five types of mask attacks. The videos are captured with the variations in pose, lighting, and expression. SiW-M contains only 27 videos of 12 subjects for the silicone mask attack. To include sufficient silicone mask attack videos in our database while focusing on prevalent 2D and 3D attacks, we have merged print, replay, and five types of 3D mask attacks from the SiW-M database with SMAD. Fig. 3 shows sample images from the above mentioned merged database.

B. Experimental Protocols

We divide the merged database into two non-overlapping parts. For each part, we define a frame-based (classifying every single frame as attack or genuine) protocol.

Intra-database Protocol: This part contains the print and the replay attack videos from SiW-M and all the silicone attack videos from SMAD. For genuine data, we use all the 65 genuine videos from SMAD and 217 videos from the train split of the genuine data in SiW-M. We perform three-fold cross-validation in our experiments. Videos from each class (genuine, print, replay, and mask) of intra-database are equally divided into three non-overlapping folds. In each iteration of

cross-validation, the model is trained on two folds and tested on the third fold. Table II summarizes the details of this protocol.

Cross and Unseen Attack Protocol: This part is used to emulate cross-database testing and it helps evaluate the performance on unseen attacks. It includes all the five 3D mask attack videos of SiW-M and 131 genuine videos from the test split of the genuine data in SiW-M. Table III presents the details of this protocol. The three trained models, each from the three iterations of cross-validation performed in the intra-database protocol, are evaluated on this part. The results are reported as the average for the three models. Testing on silicone mask attack videos of SiW-M emulates a cross-database scenario since the models are trained only on silicone mask videos from SMAD. Further, the other four 3D mask attacks: paper mask, half mask, transparent mask, and mannequin head are unseen attacks since the three trained models have not seen these attacks during training. We expect that since the training data has silicone mask attack samples, our proposed architecture should be able to generalize on these similar but unseen mask attacks.

C. Evaluation Metrics

We have used the standard evaluation metrics defined by ISO/IEC 30107-3 [30]: Receiver Operating Characteristic (ROC) curve, Attack Presentation Classification Error Rate (APCER) and Bona Fide Presentation Classification Error Rate (BPCER), and Average Classification Error Rate (ACER). **ROC** is the plot of true positive rate (TPR) vs. false positive

rate (FPR) calculated while varying the decision threshold for classification. The threshold for classification is computed on the Equal Error Rate (EER) of ROC, which is the error rate at the point where TPR equals the FPR. **APCER** is the fraction of presentation attack attempts that were successful and thus classified as genuine. **BPCER** is the fraction of bonafide samples falsely rejected as spoof. **ACER** is the average of APCER and BPCER. We report these metrics after averaging across the test sets of the three-fold cross-validation. In each iteration, we use the training fold to determine the threshold corresponding to the equal error rate (EER) and then use it to calculate ACER, APCER, and BPCER on the test sets.

V. EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS

This section summarizes the experiments and their results to demonstrate the effectiveness of the proposed architecture. We use the Face Detector in Dlib library, which is a HOG+SVM based algorithm to crop face images from the videos of SiW-M and SMAD databases. We first describe the implementation details of the four algorithms used for comparison, then show their performance in the intra-database testing followed by the cross-domain testing results and analysis. For the cross-domain setting, the proposed algorithm is also compared with existing PAD algorithm, namely auxiliary supervision [31].

A. Existing Algorithms

The proposed algorithm is compared with two texture-based algorithms: LBP+HOG and Multi-scale LBP [9] and two deep learning algorithms: ResNet50 and DenseNet121.

1) Hand-Crafted Features + SVM: In the first algorithm, we concatenate two popular hand-crafted features, namely HOG [32] (Histogram of Oriented Gradients) and LBP [33] (Local Binary Patterns) histogram from an image, then apply SVM (Support Vector Machine) for classification. For LBP histogram, we obtain uniform non-rotation invariant 59-bin histogram vector computed using 8 sampling points on a circle of radius 1. For HOG features, we use 9 orientation bins, 16×16 pixels per cell, and apply L2-Hysteresis block normalization over blocks of 3×3 cells. Therefore, the dimension of the HOG features is 324 (=$9 \times 3 \times 3 \times 2 \times 2$). In the second algorithm, we have used the formulation proposed in [9], to calculate the multi-scale LBP histogram feature vector. The final histogram vector is passed to the non-linear SVM with radial basis function kernel for classification.

2) Deep CNNs: We have used ImageNet [28] database trained CNNs, namely ResNet50 [27] and DenseNet121 [3] and fine-tuned for face PAD. The output layer of these CNNs are replaced by a fully connected softmax layer of 2 nodes representing real and attack class. The networks are fine-trained using stochastic gradient descent (SGD) to optimize categorical cross-entropy loss. The batch size and initial learning rate are set to 56 and 0.01, respectively.

B. Results for Intra-Database Protocol

The intra-database protocol described in section IV-B is followed for obtaining the results. For each algorithm, we

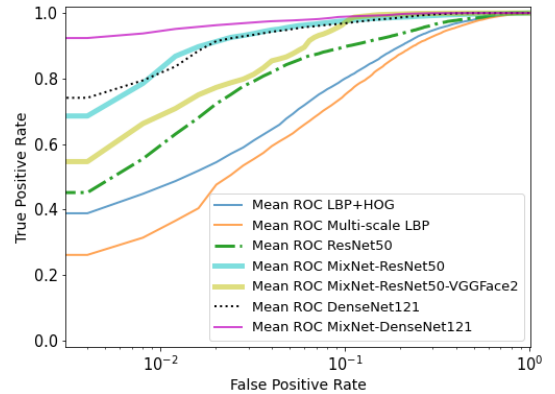| Architecture	ACER	APCER	BPCER
LBP+HOG	14.98 ± 2.90	14.99 ± 6.15	14.96 ± 4.81
Multi-scale LBP [9]	16.01 ± 1.64	12.60 ± 0.65	19.43 ± 3.06
ResNet50	10.05 ± 2.82	10.91 ± 5.21	9.18 ± 5.43
MixNet-ResNet50	6.41 ± 0.69	$\mathbf{2.34 \pm 1.34}$	10.49 ± 2.06
MixNet-ResNet50-VF2	6.85 ± 2.89	7.24 ± 4.46	$\mathbf{6.47 \pm 1.86}$
DenseNet121	$\mathbf{6.02 \pm 0.63}$	7.16 ± 2.61	$\mathbf{4.88 \pm 3.87}$
MixNet-DenseNet121	$\mathbf{4.52 \pm 0.90}$	$\mathbf{1.76 \pm 1.31}$	7.28 ± 1.61

Fig. 4. ROC for intra-database protocol.

have performed three-fold cross-validation. The error rates corresponding to intra-database protocol are reported in Table IV.

The combination of LBP and HOG yields an average error rate of 14.98%, whereas, the LBP features computed over multiple scales yields the highest error rate among all the algorithms. The algorithms based on CNN models outperform the hand-crafted features based algorithms. Among the two CNN models used, the deeper model with 121 layers shows the lowest error rates for APCER and BPCER. However, the proposed MixNet-DenseNet121 reduces the attack error (i.e., APCER) and overall error (i.e., ACER) by 75.42% and 24.92%, respectively. On the other hand, the proposed MixNet with the ResNet50 model shows an improvement of 36.22% in the average detection error rate. It is interesting to note that DenseNet121 shows the lowest BPCER value. This could be because vanilla DenseNet and ResNet only have to classify samples as genuine or not genuine. In contrast, their corresponding MixNet versions focus on detecting the three types of attack. The MixNet corresponding to ResNet model pre-trained on face images shows a slightly higher error rate than object images based counterpart. Fig. 4 shows the ROC curves for intra-database experiments obtained using different PAD algorithms.

Attack-wise APCER: We have also performed the ablation study to see whether simultaneous learning in MixNet helps in detecting a specific attack. The 3 fold cross-validation experimental results on merged database shows that the proposed MixNet yields APCER value of $0.0 \pm 0.0\%$, $0.08 \pm 0.12\%$, and

Architecture	ACER	APCER	BPCER
LBP+HOG	35.68 ± 0.99	57.70 ± 0.65	13.66 ± 1.47
Multi-scale LBP [9]	32.86 ± 0.90	51.55 ± 2.60	14.18 ± 2.03
ResNet50	35.48 ± 0.54	56.75 ± 2.32	14.21 ± 1.41
MixNet-ResNet50	$\mathbf{23.69 \pm 4.77}$	$\mathbf{35.48 \pm 10.10}$	$\mathbf{11.89 \pm 0.96}$
MixNet-ResNet50-VF2	32.95 ± 1.29	53.81 ± 3.27	12.10 ± 0.71
DenseNet121	30.84 ± 1.97	50.48 ± 4.24	$\mathbf{11.20 \pm 0.51}$
MixNet-DenseNet121	$\mathbf{24.72 \pm 0.61}$	$\mathbf{36.93 \pm 0.95}$	12.51 ± 0.64

$2.87 \pm 2.06\%$ for print, replay, and mask attack respectively. On the other hand, separately training three DenseNet121 each dedicated to classify a specific attack and taking the maximum of their output scores gives APCER of $2.94 \pm 2.03\%$, $5.74 \pm 4.22\%$, and $6.71 \pm 0.50\%$ for print, replay, and mask attack, respectively. If we take the average of output scores, the APCERs are $1.97 \pm 2.36\%$, $3.70 \pm 4.25\%$, and $7.65 \pm 3.80\%$ for print, replay, and mask attack, respectively. Finally, vanilla DenseNet121 yields APCER of $1.59 \pm 0.85\%$, $1.68 \pm 1.45\%$, and $11.08 \pm 3.46\%$ for the three attacks, respectively. The lower error rate for each attack showcases the advantage of proposed MixNet over simpler learning.

C. Results for Cross and Unseen Attack Protocol

As described in section IV-B, the models corresponding to each fold trained on intra-database protocols are used to evaluate on cross-database and unseen attack settings. Table V shows the average ACER, APCER, and BPCER for this protocol. It is found that the proposed MixNet utilizing ResNet50 performs slightly better than the MixNet utilizing the DenseNet121. The PAD algorithms based on hand-crafted features and deep CNN models yield APCER of at-least 50.48% while their BPCER is significantly lower. The proposed MixNet-ResNet50 reduces the error rates of ResNet50 by at-least 16.33%. Similarly, the MixNet-DenseNet121 reduces the error rates of the fine-tuned DenseNet121 model significantly. For example, the attack samples detection error rate of MixNet is 26.84% lower than the DenseNet121 model. Interestingly, we have observed that the MixNet with ResNet50 model pre-trained on VGGFace2 database yields higher ACER value as compared to the MixNet with ResNet50 model pre-trained on the ImageNet database.

Table VI shows the attack detection error for each of the five 3D mask attacks. The hand-crafted algorithms which lack generalizability fail significantly for silicone mask and transparent mask. The paper mask is found to be the easiest attack to be detected, which might be because it lacks the smooth texture and suffers from edge artifacts. Even in such a case, the ResNet50 model shows more than 97% error rate. On unseen attack types such as silicone mask, paper mask, half mask, and mannequin, the error rate of the proposed MixNet-DenseNet121 is 11.54%, 4.54%, and 21.56%, and 3.54%, respectively. The effectiveness of the proposed algorithm on attacks such as mannequin, which is not explored previously in the literature, shows that it is generalizable to handle the
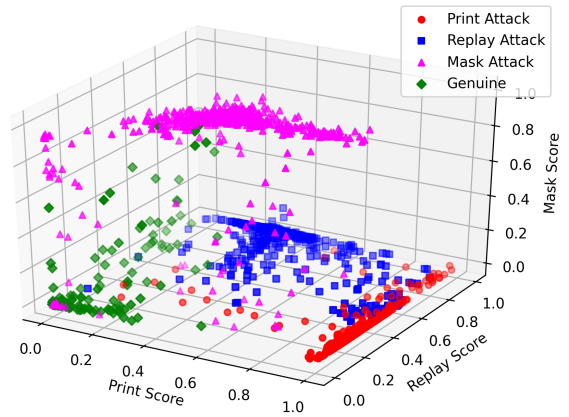

Fig. 5. Visualisation of output scores from the three sub-architectures of MixNet-DenseNet121 for a subset of test samples.

real-world scenarios. On the easiest paper-based mask, the fine-tuned DenseNet121 model shows a 26.12% error rate, which is 82.62% higher than the MixNet. On the remaining attacks, the error rate of the proposed algorithm is at-least 12.24% lower than DenseNet121. Further, the performance of the proposed algorithm for mannequin attack detection is 78.4% and 7.9% better than **auxiliary supervision** [31] and **deep tree** [5], respectively. The silicone and half-mask detection error rates of the proposed MixNet-DenseNet121 are 36.8% and 22.0% lower than the auxiliary supervision [31], respectively. We have also observed that every model performed poorly on samples from a transparent mask. It may be because the real face behind a transparent mask is almost visible, closely resembling a natural face. However, the error rate of the proposed algorithm is 27.9% lower than auxiliary supervision approach [31].

D. Visualization and Analysis

A 3D scatter plot shown in Fig. 5 is used to visualize the output scores from three sub-architectures of MixNet and to showcase their importance in detecting the particular attacks. It is observed that the samples of genuine class and the three attack classes form four separate clusters with minimal overlap. This shows that MixNet can not only detect an attack but even classify the type of the attack. It is observed that the scores of mask attack samples are intermixed with the cluster of genuine samples. In contrast, print and replay attack samples are distant from the cluster of genuine samples. It implies that mask attacks resemble real-life face texture and quality more than print and replay attacks. Thus, the major challenge lies in detecting mask attacks.

Deep Learning based techniques are often considered as black boxes. To gain an insight into what kind of features are learned by MixNet, we visualize the class activation maps (CAM) [34]. CAM is used to highlight the regions in an input image that are most effective for classification. Fig. 6 provides some interesting insights: print and replay attacks are detected using the regions around the nose and mouth, while for the

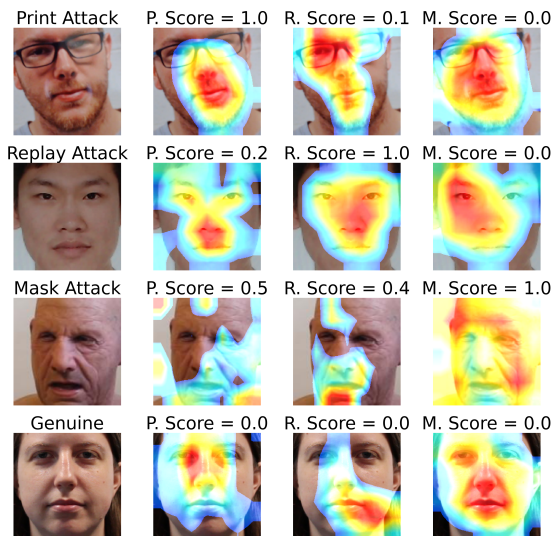| Architecture	Silicone Mask	Paper Mask	Half Mask	Transparent Mask	Mannequin
LBP+HOG	53.33	21.44	54.06	83.85	26.98
Multi-scale LBP [9]	45.68	4.84	42.91	84.74	21.03
ResNet50	**12.84**	97.50	44.28	98.18	31.32
MixNet-ResNet50	16.22	**1.00**	**26.41**	**71.54**	**4.78**
MixNet-ResNet50-VF2	17.74	10.20	61.73	82.46	23.67
DenseNet121	23.12	26.12	39.92	92.94	9.34
MixNet-DenseNet121	**11.54**	**4.54**	**21.56**	**81.56**	**3.54**

Fig. 6. The Class Activation Maps (CAMs) of four kinds of images obtained from MixNet-DenseNet121. From left to right: original image, CAM of print, replay, and mask architecture, respectively. The CAMs highlight the image regions used by each sub-architecture to detect the corresponding attack. P. Score, R. Score, M. Score represent the scores of Print, Replay, and Mask class, respectively.

mask attacks, the full face region is required. The depth around the nose region is different from the rest of the face, and hence, print and replay attacks can be detected using these regions. Mask attacks, however, can be detected using the areas where there is an opening in the eyes and mouth and also by finding discriminative patterns on cheek and forehead regions.

VI. ABLATION STUDY

In this section, we showcase the performance of the proposed algorithm on existing face PAD databases using their predefined protocol. Other than that, in place of training the three sub-architectures simultaneously as MixNet, the performance of sub-architectures trained independently is also studied.

Results on Existing Databases: We have performed experiments on the existing databases, namely Replay-Attack [6] and MSU-MFSD [7] for an extensive comparison of MixNet with other face PAD algorithms. **Replay-Attack** consists of 1200 real, print, and replay videos of 50 subjects. **MSU-MFSD** has 280 videos of photo and video attack attempts of 35 subjects. We have used the predefined train test split of both the databases to make a fair comparison with the existing algorithm. Since Replay-Attack and MSU-MFSD do not contain mask attack samples, the MixNet-DenseNet for these experiments only had two sub-architectures, one each for print and replay attack. Table VII shows the face PAD results of the proposed and existing algorithms on Replay-Attack and MSU-MFSD. The results show the effectiveness of the proposed algorithm by surpassing several existing algorithms based on the fusion of classifiers, image regions, and features. For example, the recently proposed algorithm DR-UDA [25] consisting of three modules: a source domain metric learning network (ML-Net), an unsupervised adversarial domain adaptation module (UDA-Net), and a disentangled representation learning module (DR-Net) achieves 1.3% HTER and 6.3% EER on the Replay-Attack and MSU database, respectively. On the other hand, the proposed MixNet improves the performance to 0.6% HTER and 0.4% EER on Replay-Attack and MSU-MFSD datasets, respectively.

Simultaneous vs. Independent Sub-architectures Training: We have also compared the proposed MixNet to a method where we separately train models, each dedicated to classifying a specific attack type. The final output score is optimized on the training or validation set using the maximum or average rule of these models' output scores. We selected the best model among Xception [35], DenseNet121 [3], and ResNet50 [27] for each attack type based on HTER on the validation set. For Replay-Attack, the best settings were ResNet50 for the print attack model, DenseNet121 for the replay attack model, and the average score for final output. Similarly, for MSU-MFSD, the best configuration used DenseNet121 for both the attack types and took the maximum of these scores for the final output score. As shown in Table VII, the lowest test EER on MSU-MFSD is **2.36%** which is 1.96% higher than the MixNet. Similarly, the lowest test HTER of the independent sub-architectures on Replay-Attack is **0.68%** higher than the proposed MixNet.

VII. CONCLUSION AND FUTURE WORK

The vulnerability of facial recognition algorithms to presentation attacks limit their usability for security purposes. Thus, it becomes essential to develop more reliable and robust algorithms to detect such attacks on facial recognition. This paper introduces a novel architecture termed as **MixNet**, which utilizes three sub-architectures to identify the particular presentation attack. Experimental results show that MixNet

TABLE VII
INTRA-DATABASE EVALUATION ON REPLAY-ATTACK (HTER%) AND
MSU-MFSD (EER%).

Method	Replay-Attack	MSU-MFSD
Haralick Features [10]	-	5.0
Deep Learning [36]	2.1	5.8
ResNet18 [27]	2.8	8.7
DR-UDA (ResNet18) [25]	1.4	6.0
SE-ResNet18 [37]	2.4	8.7
DR-UDA (SE-ResNet18) [25]	1.3	6.3
Multi-Regional CNN [38]	1.6	-
CCoLBP+Ensemble Learning [17]	4.0	5.0
SfSNet [39]	3.1	-
Independently Optimized Sub-Nets	1.3	2.4
Ours (MixNet-DenseNet)	**0.6**	**0.4**

outperforms multiple face PAD algorithms based on CNN architectures and hand-crafted features to detect seen and unseen attacks. Currently, for each sub-architecture in MixNet, the same network has been used. We plan to explore the selection of different architectures such that they are state of the art for detecting the corresponding attack. Finally, we believe the application of MixNet is not limited to face presentation attack detection but can also be extended to other biometrics such as iris and fingerprint PAD.

ACKNOWLEDGMENT

REFERENCES

[1] S. Jia, G. Guo, and Z. Xu, "A survey on 3d mask presentation attack detection and countermeasures," *PR*, vol. 98, p. 107032, 2020.

[2] S. Bharadwaj, T. I. Dhamecha, M. Vatsa, and R. Singh, "Computationally efficient face spoofing detection with motion magnification," in *IEEE CVPRW*, 2013, pp. 105–110.

[3] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017, pp. 2261–2269.

[4] I. Manjani, S. Tariyal, M. Vatsa, R. Singh, and A. Majumdar, "Detecting silicone mask-based presentation attack via deep dictionary learning," *IEEE TIFS*, vol. 12, no. 7, pp. 1713–1723, July 2017.

[5] Y. Liu, J. Stehouwer, A. Jourabloo, and X. Liu, "Deep tree learning for zero-shot face anti-spoofing," in *IEEE CVPR*, 2019, pp. 4680–4689.

[6] I. Chingovska, A. Anjos, and S. Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *BIOSIG*, 2012, pp. 1–7.

[7] D. Wen, H. Han, and A. K. Jain, "Face spoof detection with image distortion analysis," *IEEE TIFS*, vol. 10, no. 4, pp. 746–761, 2015.

[8] D. Gragnaniello, G. Poggi, C. Sansone, and L. Verdoliva, "An investigation of local descriptors for biometric spoofing detection," *IEEE TIFS*, vol. 10, no. 4, pp. 849–863, 2015.

[9] J. Määttä, A. Hadid, and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," in *IJCB*, 2011, pp. 1–7.

[10] A. Agarwal, R. Singh, and M. Vatsa, "Face anti-spoofing using haralick features," in *IEEE BTAS*, 2016, pp. 1–6.

[11] A. Agarwal, M. Vatsa, and R. Singh, "CHIF: Convoluted histogram image features for detecting silicone mask based face presentation attack," *IEEE BTAS*, 2019.

[12] F. Peng, L. Qin, and M. Long, "Face presentation attack detection using guided scale texture," *MTA*, vol. 77, no. 7, pp. 8883–8909, 2018.

[13] A. Agarwal, D. Yadav, N. Kohli, R. Singh, M. Vatsa, and A. Noore, "Face presentation attack with latex masks in multispectral videos," in *IEEE CVPRW*, 2017, pp. 275–283.

[14] I. Chingovska, J. Yang, Z. Lei, D. Yi, S. Z. Li, O. Kahm, C. Glaser, N. Damer, A. Kuijper, A. Nouak *et al.*, "The 2nd competition on counter measures to 2d face spoofing attacks," in *ICB*, 2013, pp. 1–6.

[15] J. Komulainen, A. Hadid, M. Pietikäinen, A. Anjos, and S. Marcel, "Complementary countermeasures for detecting scenic face spoofing attacks," in *IEEE ICB*, 2013, pp. 1–7.

[16] T. A. Siddiqui, S. Bharadwaj, T. I. Dhamecha, A. Agarwal, M. Vatsa, R. Singh, and N. Ratha, "Face anti-spoofing with multifeature videolet aggregation," in *ICPR*, 2016, pp. 1035–1040.

[17] F. Peng, L. Qin, and M. Long, "Face presentation attack detection based on chromatic co-occurrence of local binary pattern and ensemble learning," *JVCIR*, vol. 66, p. 102746, 2020.

[18] A. Agarwal, A. Sehwag, R. Singh, and M. Vatsa, "Deceiving face presentation attack detection via image transforms," in *IEEE BigMM*, 2019, pp. 373–382.

[19] A. Agarwal, A. Sehwag, M. Vatsa, and R. Singh, "Deceiving the protector: Fooling face presentation attack detection algorithms," *IEEE/IAPR ICB*, 2019.

[20] D. Menotti, G. Chiachia, A. Pinto, W. R. Schwartz, H. Pedrini, A. X. Falcao, and A. Rocha, "Deep representations for iris, face, and fingerprint spoofing detection," *IEEE TIFS*, vol. 10, no. 4, pp. 864–879, 2015.

[21] X. Tu and Y. Fang, "Ultra-deep neural network for face anti-spoofing," in *ICNNP*. Springer, 2017, pp. 686–695.

[22] S. Mehta, A. Uberoi, A. Agarwal, M. Vatsa, and R. Singh, "Crafting a panoptic face presentation attack detector," *IEEE ICB*, 2019.

[23] S. Jia, G. Guo, Z. Xu, and Q. Wang, "Face presentation attack detection in mobile scenarios: A comprehensive evaluation," *I&VC*, vol. 93, p. 103826, 2020.

[24] R. Singh, A. Agarwal, M. Singh, S. Nagpal, and M. Vatsa, "On the robustness of face recognition algorithms against attacks and bias," *AAAI*, pp. 13583–13589, 2020.

[25] G. Wang, H. Han, S. Shan, and X. Chen, "Unsupervised adversarial domain adaptation for cross-domain face presentation attack detection," *IEEE TIFS*, vol. 13, no. 7, pp. 1794–1809, 2020.

[26] S. Zhang, A. Liu, J. Wan, Y. Liang, G. Guo, S. Escalera, H. J. Escalante, and S. Z. Li, "Casia-surf: A large-scale multi-modal benchmark for face anti-spoofing," *IEEE TBIOM*, vol. 2, no. 2, pp. 182–193, 2020.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE CVPR*, 2016, pp. 770–778.

[28] J. Deng, W. Dong, R. Socher, L. Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *CVPR*, 2009, pp. 248–255.

[29] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," in *IEEE FG*, 2018, pp. 67–74.

[30] "ISO/IEC 30107-3: Information technology International Organization for Standardization. Standard, International Organization for Standardization," Feb. 2016.

[31] Y. Liu, A. Jourabloo, and X. Liu, "Learning deep models for face anti-spoofing: Binary or auxiliary supervision," in *CVPR*, 2018, pp. 389–398.

[32] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1, 2005, pp. 886–893 vol. 1.

[33] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE TPAMI*, vol. 24, no. 7, pp. 971–987, 2002.

[34] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *IEEE CVPR*, 2016, pp. 2921–2929.

[35] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *IEEE CVPR*, 2017, pp. 1251–1258.

[36] H. Li, W. Li, H. Cao, S. Wang, F. Huang, and A. C. Kot, "Unsupervised domain adaptation for face anti-spoofing," *IEEE TIFS*, vol. 13, no. 7, pp. 1794–1809, 2018.

[37] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *IEEE/CVF CVPR*, 2018, pp. 7132–7141.

[38] Y. Ma, L. Wu, Z. Li *et al.*, "A novel face presentation attack detection scheme based on multi-regional convolutional neural networks," *Pattern Recognition Letters*, vol. 131, pp. 261–267, 2020.

[39] A. Pinto, S. Goldenstein, A. Ferreira, T. Carvalho, H. Pedrini, and A. Rocha, "Leveraging shape, reflectance and albedo from shading for face presentation attack detection," *IEEE TIFS*, vol. 15, pp. 3347–3358, 2020.