DAMAD: Database, Attack, and Model Agnostic Adversarial Perturbation Detector

Akshay Agarwal, *Member, IEEE*, Gaurav Goswami, *Member, IEEE*, Mayank Vatsa, *Senior Member, IEEE*, Richa Singh, *Fellow, IEEE*, and Nalini K. Ratha, *Fellow, IEEE*

***Abstract*—Adversarial perturbations have demonstrated the vulnerabilities of deep learning algorithms to adversarial attacks. Existing adversary detection algorithms attempt to detect the singularities; however, they are in general, loss-function, database, or model dependent. To mitigate this limitation, we propose *DAMAD*—a generalized perturbation detection algorithm which is agnostic to model architecture, training data set, and loss function used during training. The proposed adversarial perturbation detection algorithm is based on the fusion of autoencoder embedding and statistical texture features extracted from convolutional neural networks. The performance of DAMAD is evaluated on the challenging scenarios of cross-database, cross-attack, and cross-architecture training and testing along with traditional evaluation of testing on the same database with known attack and model. Comparison with state-of-the-art perturbation detection algorithms showcase the effectiveness of the proposed algorithm on six databases: ImageNet, CIFAR-10, Multi-PIE, MEDS, point and shoot challenge (PaSC), and MNIST. Performance evaluation with nearly a quarter of a million adversarial and original images and comparison with recent algorithms show the effectiveness of the proposed algorithm.**

***Index Terms*—Adversarial examples, adversarial perturbation, attack agnostic, cross-attack, cross-database, cross-model, database agnostic, model agnostic.**

I. Introduction

HIGH accuracies of deep learning networks have motivated the development of automated solutions for a variety of tasks ranging from information retrieval to disease prediction, and surveillance. However, recent research efforts [11], [65] have demonstrated that the singularities of deep networks can be exploited to design attacks for corresponding networks. The widespread popularity of deep learning algorithms and their vulnerability to adversarial examples has motivated research towards detecting such attacks. Detection can act as the first crucial stage of defense against adversarial attack and the detected examples can be discarded or processed further to remove the adversarial effects for correct classification.

Perturbation detection algorithms generally assume that the model, attack, and data characteristics are known and focus on intradatabase, intraattack, and intramodel training-testing. However, in real-world settings, an attacker may be using unseen model trained on unknown database to generate perturbed samples. As shown in Fig. 1(a), this introduces three main challenges in adversarial perturbation detection that may affect the performance of perturbation detection algorithms.

1) *Cross-Database Variations:* It refers to the scenario when the perturbation detection model is trained on one database and tested on a different database. For instance, when training is performed using the CMU MultiPIE [28] database and testing is performed on the point and shoot challenge (PaSC) [7] face database.
2) *Cross-Model Variations:* It refers to the scenario when the perturbation detection model is trained on adversarial images generated from one DNN architecture while the test cases are generated from another attack DNN model. For instance, perturbed images for training are generated using VGG-16 [58] model but the test images are generated using ResNet-152 model [32].
3) *Cross-Attack Variations:* It refers to the scenario where the perturbation detection algorithm is trained on one attack and tested on another. For instance, perturbation detection is trained with l_1 loss and tested with l_2 loss-based attack.

In this research, we have developed a generalized defense algorithm termed as ***DAMAD:*** Database, Attack, and Model Agnostic Detector. In order to achieve generalizability, the proposed approach follows "ensemble of experts" or fusion approach and combines different features from multiple "experts" (algorithms) [as shown in Fig. 1(b)]. The key highlights of this research are as follows.

1) A novel adversarial perturbation detection algorithm is proposed which is an amalgamation of a nonlinear embedding obtained from an autoencoder (AE) and statistical texture attributes obtained from DenseNet feature-maps.

Manuscript received August 27, 2019; revised March 31, 2020 and August 12, 2020; accepted January 8, 2021. The work of Akshay Agarwal was supported by the Visvesvaraya Ph.D. Fellowship. The work of Richa Singh was supported in part by a research grant from MeitY, India. The work of Mayank Vatsa was supported in part by a research grant from MeitY, India, and in part by the Swarnajayanti Fellowship by the Government of India. *(Corresponding author: Mayank Vatsa.)*

Akshay Agarwal is with the Department of Computer Science Engineering, IIIT Delhi, New Delhi 110020, India (e-mail: akshaya@iiitd.ac.in).

Gaurav Goswami is with IBM Singapore, Singapore S189767 (e-mail: gauravgs@iiitd.ac.in).

Mayank Vatsa and Richa Singh are with the Department of Computer Science Engineering, IIT Jodhpur, Jheepasani 342037, India (e-mail: mvatsa@iitj.ac.in; richa@iitj.ac.in).

Nalini K. Ratha is with the Department of Computer Science and Engineering, SUNY-Buffalo, Buffalo, NY 14260 USA (e-mail: nratha@ub.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3051529.

Digital Object Identifier 10.1109/TNNLS.2021.3051529

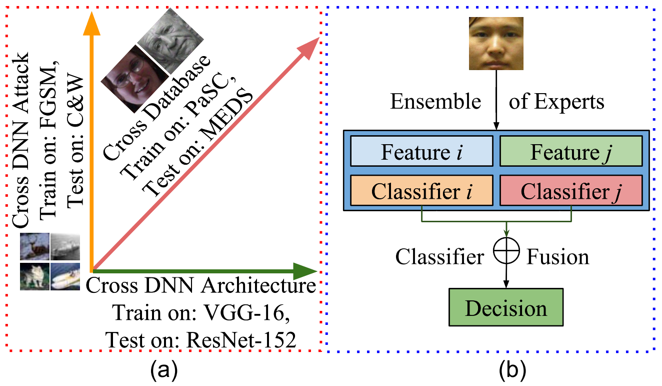


Fig. 1. (a) Three challenges in adversarial perturbation detection: 1) cross-database; 2) cross-architecture; and 3) cross-attack (i.e., cross DNN Loss). (b) Motivation toward generalized adversarial detection approach.

2) The proposed detection algorithm is evaluated on object, face, and digit recognition problems. Extensive experiments with multiple publicly available databases and deep networks demonstrate the efficacy of the algorithm in detecting different kinds of attacks.
3) Experiments pertaining to cross-database, cross-model, and cross-attack (DNN loss) scenarios demonstrate the effectiveness of DAMAD; and the strength of DAMAD is also evaluated against a white-box attack[1] and the proposed fusion approach shows resiliency against these attacks. The proposed algorithm also outperforms recent detection algorithms such as adaptive noise reduction (ANR) [42], Bayesian Uncertainty (BU) [18], CNN response approach [26], local intrinsic dimensionality (LID) [45], Base-OOD [33], ODIN [43], ESRM [44], and Mahalanobis [40] based algorithms.

To the best of our knowledge this is the first work where adversarial detection algorithm is proposed which is agnostic to multiple attack algorithms, CNN models, and databases. Based on the generalizability analysis across various unseen conditions, it is our assertion that the proposed DAMAD algorithm can be effectively used against any adversarial attacks.

II. Literature Review

Since the finding of singularities of deep learning networks, research efforts are ongoing in the direction of adversarial attack generation, and detection of adversarial examples [5]. In order to promote research in this area, several toolboxes have been developed to generate the adversarial examples using these generation algorithms and also to evaluate the effectiveness of perturbation detection algorithms [23], [51]. Recently, a detailed review paper of adversarial attack and defenses is presented by Yuan *et al.* [72] and Singh *et al.* [60].

A. Generation

The effect of adversarial perturbations on a deep learning model was first demonstrated by Szegedy *et al.* [65]. Goodfellow *et al.* [25] proposed a gradient-based algorithm for generating adversaries. The gradient is computed while training the deep neural network concerning the input while minimizing the network's loss. The gradient can be applied once or can be applied iteratively for a more robust attack. After these seminal works, several attack algorithms such as optimization-based [11], [15], gradient manipulation based, single-pixel modification [63], universal approach [49], natural filters [4], generative network-based [14], genetic algorithm-based [12], and classification boundary-based [50] attacks are proposed. Recently, Agarwal *et al.* [3] have studied the effect of the gradient to perform the attack and defense to the CNNs.

B. Detection

Existing adversarial detection algorithms can be classified into four different categories: statistical, classifier-based, dimensionality reduction-based, and image processing-based. Gong *et al.* [24], Grosse *et al.* [29], and Metzen *et al.* [48] have proposed adversarial detection based on the learning of neural network classifiers using both original and adversarial examples. Bhagoji *et al.* [8], Feinman *et al.* [18], Hendrycks and Gimpel [34], Agarwal *et al.* [2], and Li and Li [41] have presented adversarial detection either by measuring the statistical properties through a distribution or based on the measurement of dimensionality reduction techniques. Network parameter reduction based on feature squeezing is proposed by Xu *et al.* [71], which in turn helps in reducing the search space available for adversarial example generation algorithms. Lee *et al.* [40] proposed the Mahalanobis distance-based confidence score for the detection of out-of-distribution (OOD) and adversarial images detection. They assume that the features of CNN fit the class conditional Gaussian distribution. Ma *et al.* [45] analyzed the properties of an adversarial region using LID features and used it for the detection of adversarial examples. Chen *et al.* [13] proposed the defense based on focusing the attention on critical regions and contour of the image objects. Goel *et al.* [20]–[22] have presented various defense mechanisms based on the reconfiguration of the CNN architectures.

Although many detection methodologies have been proposed in the literature, a common and significant limitation is the ineffectiveness [9] in detecting challenging adversarial examples such as generated using C&W's (l_2) attack [11]. For instance, the detection algorithm proposed by Li *et al.* [41] is able to achieve perturbation detection accuracy of only 8% when used to detect adversarial samples generated using Carlini and Wagner (C&W's) attack (l_2) on the MNIST database. Similarly, on the CIFAR-10 database only 1% adversarial examples are successfully detected [9], [11]. Other defense algorithms based on gradient hiding [53], [56], input transformations [30], [70], generative networks [62], CNN-based classifiers [24], [29], [48], and single classifier [47] are also proven ineffective across stronger attacks [6], [9], [10]. Recent algorithms [54], [61] are able to provide certified defense for small perturbations on the MNIST attack database but they are not effective [67] against multiple attacks and databases.

[1] In adversarial attack research, white-box attack refers to the condition in which an attacker has access to the defense/detection mechanism and classifier. On the other hand, black-box attacks are defined where an attacker does not have access to both classifier and defense mechanism.

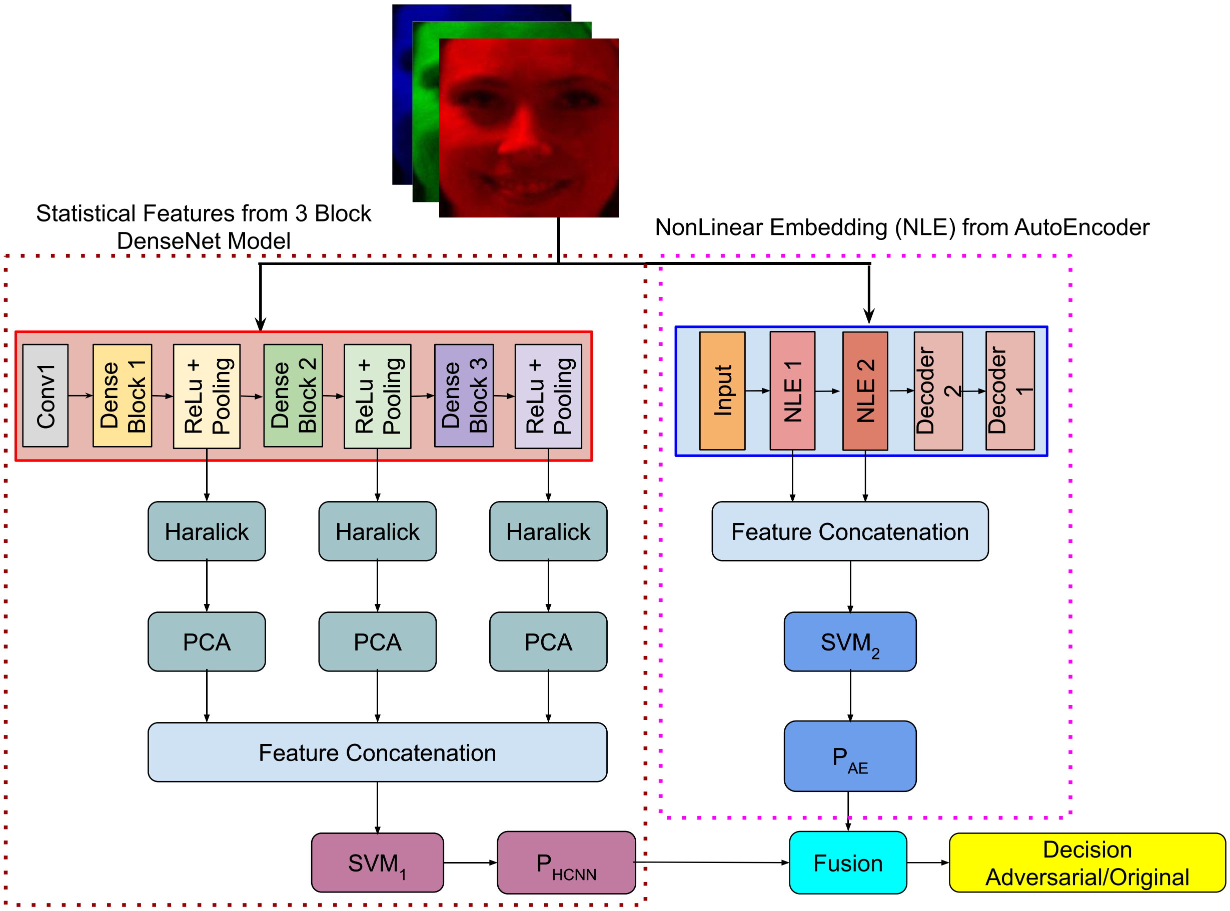


Fig. 2. Proposed *DAMAD* adversarial perturbation detection algorithm combines statistical texture attributes obtained from DenseNet-121 feature maps and AE embedding.

Not directly related to adversarial attacks, Tao and Cao [66] present the resilient learning against erroneous database labels through noisy labels.

III. Proposed Algorithm

The adversarial image generation approaches embed an imperceptible "adversarial noise" in the original image. Across different attacks, we observed that the attack algorithms differ in the kind of noise (e.g., gradient-based or magnitude), magnitude of noise (say single step or iterative), and the region where it is embedded (e.g., every pixel or salient regions only). This is similar to the watermarking and steganography literature, where the watermark or the message is embedded in the source image. This has been supported by Goodfellow *et al.* [25], where they have mentioned that adversarial perturbations can be treated as "accidental steganography."

Based on the literature and limitations discussed in Section II, we hypothesize that a single algorithm may not be able to detect different kinds of adversarial noise. Inspired from the multimodal biometrics research [55], [59] and generalized amalgamation technique for adversarial example detection [73], we postulate that an "ensemble of experts" or multiclassifier fusion approach, which combines complementary features obtained from distinct sources, can alleviate the limitations of single feature classification approaches. In other words, the multiclassifier fusion approach can better model the variabilities in original and adversarial noise classes to provide better generalizability. Furthermore, statistical features such as Haralick can help in determining the differences between "original" and "perturbed" images. Based on these assertions, we propose *DAMAD* to detect the presence of adversarial attack in an image. Fig. 2 shows the block diagram of the proposed algorithm. In the proposed algorithm, the following holds.

1) Statistical Haralick features from the intermediate layers of DenseNet are extracted and probability confidence scores from support vector machine (SVM) are computed.
2) Features from intermediate layers of an AE are extracted and SVM probability confidence scores are computed.
3) The two probability scores are combined to obtain a final decision.

A. Statistical Features From DenseNet

Goswami *et al.* [26], [27] have demonstrated that filter responses of original and adversarial examples have significant differences, i.e., CNN filters are sensitive toward adversarial noise. From this observation, we propose to use intermediate layers of a deep network to learn the differences between original and attacked samples. In place of standard CNN, this research uses DenseNet which has stronger feature propagation and substantially fewer parameters. Furthermore, we extract statistical features from the intermediate feature maps using Haralick features [31]. The Haralick features used are: angular second moment, contrast, correlation, sum of squares: variance, inverse difference moment, sum average, sum variance, sum entropy, entropy, information measure of correlation 2, difference entropy, information measure of correlation 1, and difference variance. It encodes statistical properties of an input signal and extracts global attributes such as context, correlation, and entropy. Agarwal *et al.* [1] have proposed a combination of wavelet and Haralick for the detection of face presentation attack. However, the generalizability against unseen attack is a concern of the algorithm.

The proposed *DAMAD* utilizes three blocks of DenseNet-121 CNN model [36] to learn the filters that can accentuate the differences between the set of original and perturbed images. The dense blocks are initialized from the weights of the DenseNet-121 model trained on ILSVRC [17]. Dense block 1 consists of six dense layers, whereas, blocks 2 and 3 consist of 12 and 24 dense layers, respectively. Each dense layer has two convolution layers with filter size 1×1 and 3×3. The convolution block contains batch normalization, ReLU nonlinearity followed by a convolution operation. The feature maps at any layer are the concatenation of all feature maps computed before that layer. After each dense block pooling, the transition layer is used to reduce the size of the feature maps which also helps in equating the size of each feature map. These new connectivities between the layers help in better encoding the patterns present in the input data and therefore motivated this research to compute the statistical features over the maps computed from DenseNet.

As shown in Fig. 2, in the first convolution layer, a three-channel RGB image is convolved with 12 filters followed by the first block of DenseNet model. Before passing the output to the next DenseNet block, ReLU activation and 2×2 spatial pooling are applied to the feature maps. A similar process is repeated for the next two blocks of DenseNet and filters are learned by using a fully connected (FC) layer (with two class classification). Once the network is trained, FC layer is removed and the filtered outputs at-the-end of each ReLU + Pooling is used to compute the Haralick features, i.e., 13 Haralick feature vector is computed for each filtered output in ReLU + Pooling layer. If the number of filtered output in any ReLU + Pooling layer is n, then the size of the Haralick feature vector is $13\times n$. To reduce the dimensionality, principal component analysis (PCA) is applied [68] and 99% Eigen energy is preserved. The reduced dimensional feature is then combined and classified into 2-classes (*original* and *perturbed*) using a linear SVM classifier (SVM$_1$ in Fig. 2).

B. AE Model for Perturbation Detection

As the second classifier, a denoising AE model is trained which can help discriminate between perturbed and original images. An AE captures the intrinsic properties of data and learns to abstract image properties by learning the latent space representation. Visualization of hidden layer encoding of an AE (original and adversarial images), as shown in Fig. 3, shows different spatial distribution of both classes in a nonlinear space. This property is explored for detecting adversarial noise in the input images. An unsupervised AE has a reconstruction loss function as

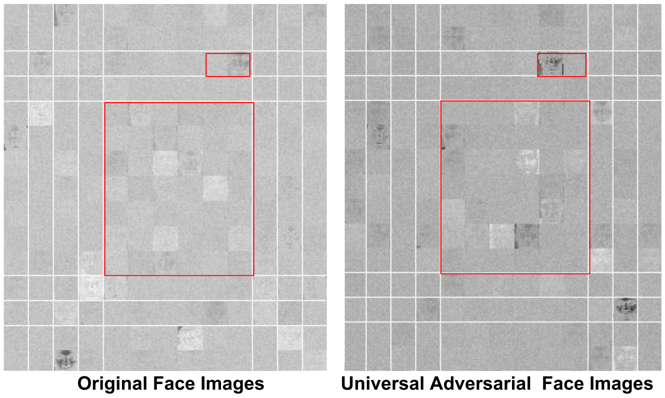


Fig. 3. Hidden layer visualization of AE. The embeddings learned on adversarial examples are more noisy as compared to clean images which might help in detecting the attack.

$$\text{argmin}_{W,W'} \; ||X - W'\phi(WX)||_2^2 + \lambda R \tag{1}$$

where W and W' are the encoding and decoding weights, ϕ is the nonlinear activation (e.g., ReLU), λ is the regularization constant, and R is the regularizer (e.g., $||.||$ norm and dropout). The stacked AE extends (1) to

$$\begin{aligned} &\text{argmin}_{W_1,\ldots,W_n,W'_1,\ldots,W'_n} \; ||X - g \circ f(X)||_2^2 + \lambda R \\ &g = W'_1\phi(W'_2 \ldots \phi(W'_n f(X))) \\ &f = \phi(W_n \cdots \phi(W_1(X))). \end{aligned} \tag{2}$$

With two encoding layers, the feature can be represented as $H_x = \phi(W_2\phi(W_1 X))$. Given an input original image X and a perturbed image Y, the features H_x and H_y are fed into a 2-class SVM [16] with Radial Basis Function kernel to distinguish between *original* and *perturbed* classes (SVM$_2$ in Fig. 2).

C. Multiclassifier Fusion

The two feature networks, Haralick features from DenseNet and AE, are combined using a late fusion approach. The classification probability scores obtained from two SVM classifiers (P_{AE} and P_{HCNN}) are combined using *sum* rule, i.e., $P_{\text{fusion}} = ((P_{\text{AE}} + P_{\text{HCNN}})/2)$ and the fused score is used to classify an input image as original or perturbed.

D. Implementation Details

The *DAMAD* algorithm is implemented in Theano environment with K40 GPU and Adam optimizer. For the AE model, given an input image of size N, two hidden layers are of size $[(N/2), (N/2)]$. For the CNN model, 12 filters of size 3×3 are used in the first convolutional layer followed by three DenseNet blocks along with 2 × 2 maxpooling layer. Furthermore, for both the models, the learning rate is 0.0001, the dropout rate is set to 0.5, and the number of epochs is 500. Geometric transformation (1°–3° rotation) and reflection of the image is used for data augmentation.

IV. Databases and Evaluation Protocol

This section summarizes the databases and attacks considered for evaluation along with the experimental protocol and existing algorithms for comparison.

A. Attacks

To evaluate the performance and generalizability of the proposed DAMAD algorithm, we have performed the experiments with different attacks: optimization based [elastic net (EN)] [15], C&W [11]), universal perturbations [49], PGD [46], gradient-based algorithms [25], [38], and DeepFool [50]. Table I provides the number of images generated for each of the databases and the adversarial model used for image generation.

B. Universal Attack or Image Agnostic Attacks [49]

We have used three different deep neural network models to generate the universal perturbed images. The DNN models used are: VGG-16 [58], ResNet-152 [32], and GoogLeNet [64]. Since these are among the best performing networks for tasks such as object recognition and face recognition, we have selected these networks to generate the universal adversary on the face and ImageNet databases.

C. DNN Loss-Based Adversarial Perturbations

Nine different types of DNN loss-based attacks are selected to generate the adversarial images from the MNIST and CIFAR-10 databases. The selected attack generation algorithms are among the most challenging attacks [9]. Gradient-based adversarial example generation algorithms are the most common in the literature and hence they are also utilized to generate adversarial images. In this research, we have used a basic version of a gradient-based algorithm known as the fast gradient sign method (FGSM) and an iterative version of FGSM (IFGSM). PGD is another stronger variant of FGSM attack which iteratively computes the adversarial noise. It is also considered the universal adversary among first-order adversaries. Other than the basic version, l_1, l_2, and EN norm minimization-based variants are also used to generate the adversarial examples.

D. DeepFool

The minimal norm perturbation is computed iteratively. The algorithms start with a clean image that resides in the decision boundary defined by the classifier. At each iteration subtle noise vector is added to the input image with the aim to take the image outside the decision boundary.

E. Attack Parameters

For C&W and EN attacks, regularization parameter (initially $c = 0.001$) is searched over nine binary steps where each step runs for 1000 iterations. The initial learning rate is set to 0.01. ADAM optimizer, and projected FISTA with square-root decaying rate are used for C&W and EN, respectively. Similarly, for IFGSM and its variants, CleaverHans[2] package is used. The best distortion parameter is selected using the fine-grained search. Ten FGM iterations are implemented with distortion parameter $\epsilon/10$ in each iteration. All other settings are kept as default for all the attacks. The experimental parameters used for adversarial examples generation are reported in Tables II and III. The original codes provided by the authors of Universal, DeepFool, and PGD attacks are used with quasi-imperceptible adversarial noise. The adversarial examples selected contain the lowest distortion.

F. Databases

The results are demonstrated with six popularly used face, object, and digit recognition databases. The face databases are: PaSC [7], CMU Multi-PIE [28], and the Multiple Encounters Data Set (MEDS) [19]. From these three databases, more than 9500 frontal or nearly frontal images are randomly selected. The object recognition databases used are CIFAR-10 [37] and ImageNet (i.e., ILSVRC-2012) [17]. We have selected 5000 images from the ImageNet database and 9000 images from the CIFAR-10 database. MNIST database [39] contains images of handwritten digits from 0 to 9. Utilizing the code provided by Chen *et al.* [15], we have selected 9000 images from the MNIST database. In total, we have more than 32 500 original images pertaining to more 2150 classes, across these six databases.

We next created adversarially perturbed images corresponding to the adversarial attacks discussed above. In total, there are more than 29 000 perturbed face images and 177 000 perturbed images from the other three databases. It is to be noted that the codes and models for adversarial generation are taken from original papers in order to avoid any bias.

G. Protocol

The evaluation protocol includes both positive and negative attack detection (i.e., original and perturbed images). The experiments are segregated according to intravariations and cross-variations (architecture/attack/database). For all the scenarios related to *intradatabase* (such as training and testing on MEDS) and *intraattack* (such as l_1-l_1) experiments, 50% of the data from both classes is randomly selected for training and the remaining 50% for testing. In *cross-database* (such as MEDS-PaSC) and *cross-attack* (such as l_1-l_2) scenarios, original/adversarial images of one database/attack are used for training while original/adversarial images of another database/attack are used for evaluation. Similarly, in the case of *"cross DNN architecture,"* adversarial images generated using one DNN model (such as VGG-16) are used to train the classifier, while at the time of testing, adversarial images

[2] https://github.com/tensorflow/cleverhans

TABLE I

NUMBER OF ORIGINAL IMAGES AND ADVERSARIAL (PERTURBED) IMAGES GENERATED FOR EACH DATABASE

Database		**Original**	**Adversarial Model**	**Perturbed**	**Classes**
Face	MEDS	836	1. DeepFool: VGG-16 2. Universal: VGG-16, GoogLeNet, and ResNet-152	2,508	518
	PaSC	7,443		22,329	293
	Multi-PIE	1,680		5,040	336
Object	ILSVRC 2012	5,000	Universal: VGG-16, GoogLeNet, and ResNet-152	15,000	1000+
	CIFAR-10	9,000	FGSM-l_1, IFGSM-l_1, FGSM-l_2, IFGSM-l_2, FGSM, IFGSM, DNN Loss (l_1, l_2, EN), PGD	100,000	10
Digit	MNIST	9,000		100,000	10

TABLE II

RANGE AND RESOLUTION OF DISTORTION PARAMETERS TO GENERATE IFGSM ADVERSARIAL SAMPLES

Method	**Grid Search**	
	Range	Resolution
IFGSM-L_∞	[10^{-3}, 1]	10^{-3}
IFGSM-L_1	[1, 10^{3}]	1
IFGSM-L_2	[10^{-2}, 10]	10^{-2}

TABLE III

EXPERIMENTAL SETUP OF C&W, EN, AND L_1 ATTACKS

Parameter	**Value**
Initial Learning Rate	0.01
Iterations	1,000
Initial Regularization	0.001
Steps	9
Optimizers	ADAM and projected FISTA

generated using another model (such as GoogLeNet) are used. We have performed evaluations with twofold unseen training-testing as well. For instance, *"cross DNN architecture and cross-database,"* where not only DNN architecture from which universal adversarial images are generated is different but the testing database is also different. It is to be noted that this is the first work to report results on "cross" training-testing conditions in three areas: cross-database, different DNN architectures, and different loss functions (l_1/l_2/$l_1 + l_2$/GSM).

H. Evaluation Metric

The results are reported using the average detection accuracy of real and adversarial examples. The detection accuracy is the average of true positive rate (TPR) and true negative rate (TNR). TPR is defined as the rate of real examples being classified as real and TNR is defined as the rate of adversarial examples being classified as adversarial. In order to maintain the class balance, in each experiment, we have used an equal number of real and adversarial examples.

I. Algorithms for Comparison

The performance of *DAMAD* is compared with three recently proposed detection algorithms: ANR [42], BU [18], Base-OOD [33], ODIN [43], ESRM [44], and CNN response [26]. The Base-OOD and ODIN use the softmax probabilities of the DNN model to identify OOD samples. The ODIN, an enhanced version of Base-OOD, uses the temperature scaling to softmax probabilities [35] and input perturbation to enlarge the softmax score gap between in-and-out distribution samples. ESRM uses the concept of steganalysis for the detection of adversarial attacks. It models the dependence between the adjacent pixels using a hidden Markov model. Other than the existing adversarial detection algorithms, *DAMAD* is compared with two deep learning models: VGG-16 [58] and DenseNet [36]. The VGG-16 and DenseNet model (pretrained on Imagenet) are fine-tuned using the adversarial and original images for perturbation detection. Along with these, detailed analysis is performed with individual components of *DAMAD*, redundant discrete wavelet transform (RDWT) + Haralick, and local binary pattern (LBP) features. The SVM classifier is trained on the training set corresponding to each protocol and detection results are reported using features computed on the testing set. A comparison with recently proposed LID [45] and Mahalanobis [40] algorithms on complex l_2 attack on CIFAR10 database is also reported.

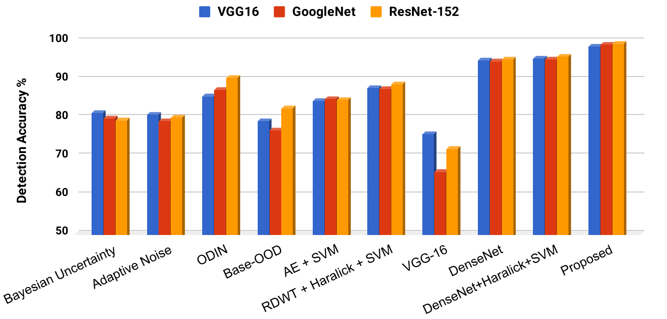


Fig. 4. Detection performance of DAMAD and existing adversary detection algorithms on the ImageNet database with universal adversarial perturbation. (Best viewed in zoom and color.)

V. RESULTS AND ANALYSIS

The results are divided into three parts. First, the results are analyzed with respect to the intravariations in database, model, and attack, followed by intervariations. Finally, the general observations made across the intravariations and intervariations experiments are discussed.

A. Results With Intravariations

Figs. 4 and 5 summarize the results on the ImageNet and the face databases in the intravariations setting. On the ImageNet and face databases with Universal attack, the proposed DAMAD correctly classifies more than 98% samples,

TABLE IV

ADVERSARIAL DETECTION PERFORMANCE OF THE PROPOSED DAMAD AND EXISTING ALGORITHMS ON THE CIFAR-10 AND MNIST DATABASES

Databases	Algorithms	Attacks									
		l_1	l_2	EN	PGD	FGSM	FGSM-l_1	FGSM-l_2	IFGSM	IFGSM-l_1	IFGSM-l_2
MNIST	Bayesian Uncertainty [18]	77.3	78.5	78.1	74.6	83.7	82.9	81.2	85.3	84.9	84.4
	Adaptive Noise Reduction [42]	78.6	79.2	79.6	77.8	82.9	82.7	82.1	85.9	85.7	85.1
	Base-OOD [33]	72.5	68.9	65.6	66.0	87.8	82.3	78.5	88.3	82.9	80.0
	ODIN [43]	78.7	72.0	75.6	73.3	88.8	86.9	84.5	90.2	86.2	84.1
	RDWT + Haralick + SVM [1]	73.4	71.6	68.0	71.2	90.3	98.9	96.8	97.9	99.9	98.8
	Proposed DAMAD	**99.1**	**99.6**	**99.5**	**99.3**	**99.8**	**100**	**100**	**100**	**100**	**100**
CIFAR-10	Bayesian Uncertainty [18]	56.1	57.3	58.6	56.5	84.0	84.4	83.5	86.8	87.7	88.1
	Adaptive Noise Reduction [42]	55.9	57.8	57.2	59.2	83.2	83.5	83.8	87.1	88.3	88.5
	Base-OOD [33]	62.2	64.6	61.0	63.1	81.8	80.3	76.3	80.3	77.7	75.0
	ODIN [43]	64.3	62.9	63.6	65.7	82.1	81.6	82.0	86.7	86.0	82.5
	RDWT+Haralick+SVM [1]	61.7	62.3	61.3	60.4	62.9	57.8	56.1	72.2	64.0	62.8
	Proposed DAMAD	**98.3**	**98.1**	**99.0**	**97.8**	**97.4**	**97.5**	**97.5**	**97.1**	**97.3**	**97.5**

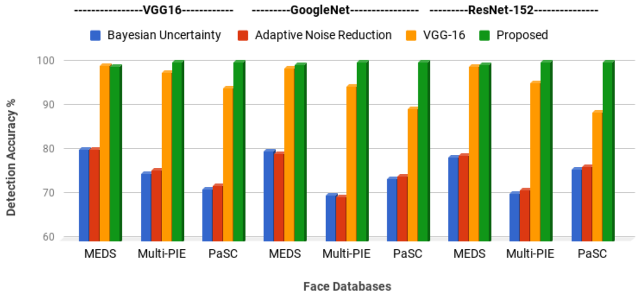


Fig. 5. Detection performance of DAMAD and existing adversary detection algorithms on face databases with universal adversarial perturbation. (Best viewed in zoom and color.)

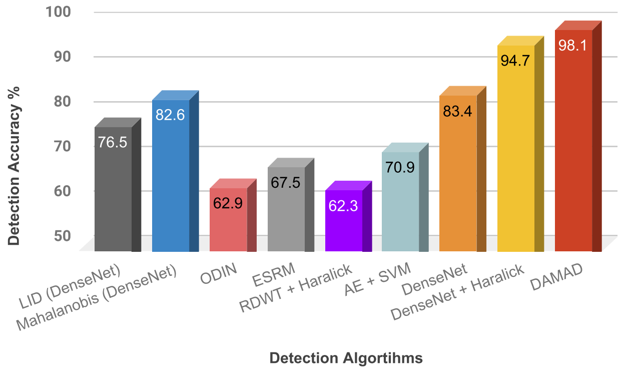
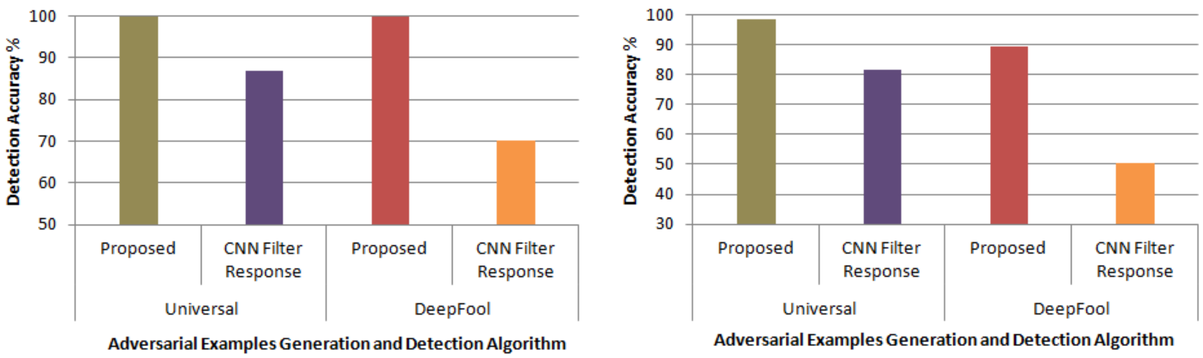


Fig. 6. Results of DAMAD and state-of-the-art detection algorithms (LID [45], Mahalanobis [40], ODIN [43], and ESRM [44]) with complex l_2 [11] attack on the CIFAR-10 [37] database.

irrespective of the models used (VGG-16, GoogLeNet, and ResNet-152). The comparative results documented in Fig. 4 show that the detection results of existing algorithms yield significantly lower performance. The performance of the detection algorithm proposed by Goswami *et al.* [26], which utilizes the intermediate filter response of VGG, is 13.2% and 16.8% lower than DAMAD on PaSC and MEDS databases, respectively. For C&W l_2 and PGD with ($\epsilon = 0.03$) attacks, the proposed DAMAD yields at-least 91% and 93% detection accuracy on face databases (MEDS, PaSC, and Multi-PIE). On the ImageNet database, in comparison to existing algorithms, there is a difference of at least 17% for all three models. On ImageNet, when VGG-16 fine-tuned model is used for universal adversarial sample detection, the accuracy is in the range of 65%–75% which is significantly lower than DAMAD. Similarly, DenseNet only-based detection model shows at-least 20% lower accuracy compared to the proposed DAMAD.

Table IV summarizes the results on the MNIST and CIFAR-10 databases with different kinds of adversarial attacks. *DAMAD* yields more than 97.1% detection accuracy on optimization and gradient-based attacks on the CIFAR-10 database. The detection performance of two existing algorithms (ANR [42] and BU [18]) on gradient-based attacks are in the range of 83.2%–88.5% on CIFAR-10 database, which is at-least 8.6% lower than *DAMAD*. On the CIFAR-10 database, *DAMAD* is at least 39.5% higher compared to existing algorithms on challenging l_1, l_2, and EN attacks. Similarly, on the MNIST database, the adversarial detection accuracy of *DAMAD* on gradient-based attacks is in the range of 99%–100%, whereas the performance of two existing algorithms are in between 81.2% and 85.9%. The *DAMAD* improves the C&W l_2 attack detection performance of LID from 76.5% to 98.1% when ResNet model is used for LID [45]. Similarly, the detection performance of Mahalanobis [40] and ESRM [44] improves at least by 6.3% and 30.6%, respectively, when the proposed DAMAD is used for complex optimization based adversarial examples detection. The results are shown in Fig. 6.

The performance is also evaluated on DeepFool adversary [50]. More than 9500 DeepFool adversarial images are generated from three face databases (MEDS, Multi-PIE, and PaSC) using the VGG-16 DNN architecture. Results of *DAMAD*, both in "intra" and "cross" database scenarios, are reported in Table V. The detection performance of DAMAD algorithm is at least 4.6% and 15.6% better than DenseNet-based classification model when trained using PaSC and Multi-PIE databases, respectively. The proposed algorithm outperforms the recently proposed algorithm based on CNN filter responses [26]. The detection accuracy of DAMAD is up to 29.8%, 39.3%, and 41.9% higher than the CNN filter



(a)

(b)

Fig. 7. Comparison of the proposed algorithm with state-of-the-art detection algorithm (*CNN filter response* [26]) on Universal [49] and DeepFool [50] attack on face databases. (Best viewed in color.) (a) Results on the PaSC database [7]. (b) Results on MEDS database [19].

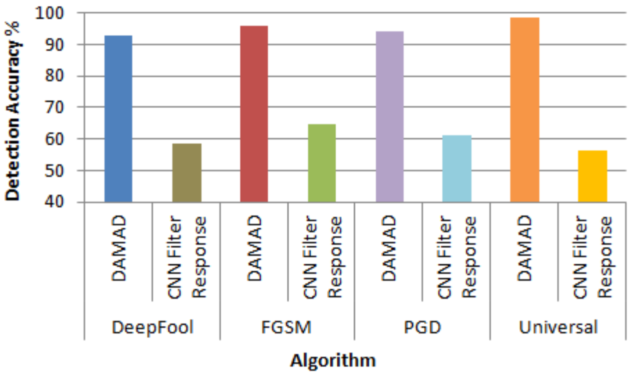


Fig. 8. Comparison of the proposed algorithm with state-of-the-art detection algorithm (CNN filter response [26]) on CIFAR-10 database. (Best viewed in color.)

TABLE V

DETECTION PERFORMANCE OF *DAMAD* FOR DEEPFOOL ADVERSARY ON FACE DATABASES WITH INTRA AND CROSS DATABASE TESTING

Train	Test		
	MEDS	Multi-PIE	PaSC
MEDS	90.2	87.6	86.2
Multi-PIE	89.5	95.0	100.0
PaSC	90.7	96.3	100.0

response algorithm on PaSC, MEDS, and CIFAR-10 databases, respectively. The results are shown in Figs. 7 and 8.[3]

1) Ablation Study: We next perform an ablation study and evaluate the effectiveness of individual steps of the algorithm on the ImageNet database. As shown in Fig. 4, it is observed that individual components such as (AE+SVM) and (DenseNet+Haralick+SVM) are individually not effective. The combination of these components in the DAMAD algorithm yields the best results. Furthermore, in place of DenseNet, RDWT+Haralick with SVM yields lower performance. This shows that all the components of the DAMAD algorithm are important for providing consistently accurate detection results across different models. Similar observations are noted for the three face databases in cross-database experiments (Table VI). The effectiveness of DenseNet over ResNet as discussed earlier is also demonstrated through experiments. On the ImageNet database, the accuracy of the DenseNet model is at-least 94.7% across all three universal perturbation generation CNN architecture. On the other hand, the performance of the ResNet-152 model is at-least 12% lower than DenseNet-121.

B. Results With Intervariations

The next set of experiments are performed to test the generalizability of the proposed algorithm with variations in testing model, attack, and database, compared to the ones for training. The combination of attacks and databases are selected according to the research in literature. For instance, DNN loss based attacks have been performed on MNIST and CIFAR, while Universal attack with different models is demonstrated for ImageNet and all three face databases.

1) Cross-Database Evaluation: Table VI summarizes the results of cross-database testing of existing algorithms, *DAMAD*, and components of *DAMAD*. The detection performance of BU and ANR is in the range of 62.8%–79.3% across different combinations of training and testing, whereas *DAMAD* lies in between 97.4% and 100%, thus demonstrating the generalization capability of the algorithm. On universal adversarial images generated using the VGG-16 model, the CNN response [26] algorithm yields 53.4% and 63.2% detection accuracies on the PaSC and MEDS databases, respectively, which are 36.8% and 45.0% lower than the *DAMAD*, respectively. We have also used the DenseNet [36] as an adversarial model and generated the adversarial examples using face databases. We have generated the universal noise vector with different fooling rates (40%, 60%, and 80%) with input variation set to 0.03. When the DenseNet adversarial model is used, the proposed defense performed similarly to other models such as VGG-16 and GoogLeNet. The accuracy ranges from 96.8% to 100% under cross-database scenarios

[3] CNN filter response [26] approach originally uses VGG network for detecting adversarial perturbations. We have performed additional experiments to understand if the performance of the CNN filter response approach is improved when DenseNet is used in place of VGG. In our experiments, we have observed that the CNN filter response approach yields up to 3% higher performance when DenseNet features are used in place of VGG features. However, it is still significantly lower compared to the proposed DAMAD.

TABLE VI

DETECTION PERFORMANCE ON FACE DATABASES WITH CROSS DATABASE TRAINING-TESTING FOR THE UNIVERSAL ATTACK. - REPRESENTS THE INTRA DATABASE CONDITIONS AND CORRESPONDING RESULTS ARE REPORTED IN FIG. 5

DNN Model	Detection Algorithm	VGG-16			GoogLeNet			ResNet-152		
Train DB		MEDS	Multi-PIE	PaSC	MEDS	Multi-PIE	PaSC	MEDS	Multi-PIE	PaSC
MEDS	Bayesian Uncertainty [18]	–	79.3	78.5	–	73.1	74.1	–	72.9	75.0
	Adaptive Noise Reduction [42]	–	77.4	79.1	–	70.5	71.8	–	71.3	72.9
	Base-OOD [33]	–	82.3	81.5	–	76.4	78.1	–	74.0	77.9
	ODIN [43]	–	84.7	82.6	–	80.0	81.2	–	77.5	78.2
	VGG-16	–	50.0	50.7	–	50.0	50.5	–	50.5	52.0
	AE + SVM	–	78.6	79.1	–	71.9	72.2	–	73.4	73.8
	DenseNet	–	97.1	96.3	–	98.6	98.5	–	98.7	99.5
	DenseNet + Haralick + SVM	–	99.4	99.2	–	99.0	99.1	–	99.2	99.1
	RDWT + Haralick + SVM [1]	–	99.7	99.6	–	97.5	96.5	–	98.0	99.3
	Proposed DAMAD	–	**100**	**100**		**100**	**99.8**		**100**	**99.9**
Multi-PIE	Bayesian Uncertainty [18]	70.1	–	72.6	70.9	–	71.0	72.8	–	70.2
	Adaptive Noise Reduction [42]	70.9	–	71.7	70.3	–	69.1	70.5	–	69.0
	Base-OOD [33]	71.9	–	73.1	69.4	–	72.3	71.0	–	74.9
	ODIN [43]	73.3	–	75.1	68.0	–	69.9	70.5	–	76.6
	VGG-16	75.3	–	79.0	82.6	–	75.6	84.3	–	75.8
	AE + SVM	73.6	–	75.3	71.9	–	71.3	72.7	–	71.7
	DenseNet	96.9	–	92.8	93.1	–	98.0	91.2	–	98.9
	DenseNet + Haralick + SVM	97.0	–	94.4	93.8	–	99.5	92.1	–	99.7
	RDWT + Haralick + SVM [1]	71.3	–	99.9	70.9	–	99.9	71.4	–	99.9
	Proposed DAMAD	**98.4**	–	**100**	**99.0**	–	**100**	**99.5**		**100**
PaSC	Bayesian Uncertainty [18]	63.9	65.3	–	62.8	63.1	–	65.2	66.9	–
	Adaptive Noise Reduction [42]	65.3	67.8	–	63.2	63.0	–	66.8	68.7	–
	Base-OOD [33]	62.1	69.8	–	64.3	66.4	–	70.1	72.3	–
	ODIN [43]	64.3	77.7	–	68.2	72.3	–	78.9	74.9	–
	VGG-16	76.9	81.3	–	68.2	75.5	–	74.4	64.0	–
	AE + SVM	70.6	71.3	–	69.7	68.4	–	70.3	71.7	–
	DenseNet	90.3	90.7	–	85.9	87.4	–	88.6	89.1	–
	DenseNet + Haralick + SVM	91.5	92.9	–	88.1	90.8	–	90.1	90.6	–
	RDWT + Haralick + SVM [1]	68.7	100	–	67.9	100	–	68.4	100	–
	Proposed DAMAD	**98.2**	**100**	–	**97.4**	**100**	–	**98.8**	**100**	

which is similar to VGG-16, GoogLeNet, and ResNet-152 model reported in Table VI. Analyzing the performance of individual components (ablation study) of the algorithm shows that each component of the algorithm is important for high detection performance, and removing any component significantly reduces the performance on some cross train-test pairs.

2) Cross-Attack Evaluation: In the cross-attack experiment, the training and testing adversarial images are generated using different attack types. Fig. 9 shows the findings related to the cross-attack situation. The average (± standard deviation) adversary detection performances on cross attack scenario are $99.2 \pm 0.6\%$, $64.2 \pm 4.1\%$, $62.3 \pm 4.3\%$, $70.3 \pm 3.3\%$, $75.4 \pm 2.8\%$, and 68.7 ± 2.7 on the MNIST database using *DAMAD*, ANR, BU, Base-OOD [33], ODIN [43], and ESRM [44] algorithms, respectively. Similarly, on the CIFAR-10 database *DAMAD*, ANR, BU, Base-OOD [33], ODIN [43], and ESRM [44] algorithms yield an average detection accuracy of 93.7 ± 1.2%, 46.7 ± 3.1%, 47.5 ± 3.2%, 56.8 ± 2.2%, 58.9 ± 1.5%, and 59.1 ± 1.9, respectively. These results show the generalizability and transferability properties of the proposed algorithm.

3) Cross-Databases and Cross-DNN-Architectures: To evaluate the generalizability in presence of more "unknown" factors, we performed another experiment as 'cross-database' and "cross-architecture." This experiment is performed using MEDS, Multi-PIE, and PaSC databases with VGG-16, GoogLeNet, and ResNet-152 architectures, where one database and one architecture is used for training while the other databases and architectures are used for testing. The proposed *DAMAD* achieves at least 99.97% accuracy which is significantly higher than the existing algorithms (less than 50% accurate). This experiment showcases that *DAMAD* is generalizable even in the case of both cross-database and cross-architecture scenarios.

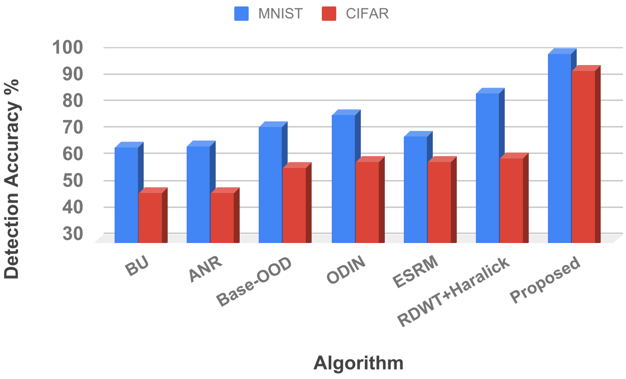


Fig. 9. Attack detection results when the model is trained on one attack and tested with other attacks. Tenfold experiments are performed, each using only one attack for training and the remaining nine attacks for testing. The average detection accuracy is reported. The comparison with BU [18], ANR [42], Base-OOD [33], ODIN [43], ESRM [44], and RDWT+Haralick [1] algorithms is also reported.

C. Discussion

We have made the following observations across different experiments.

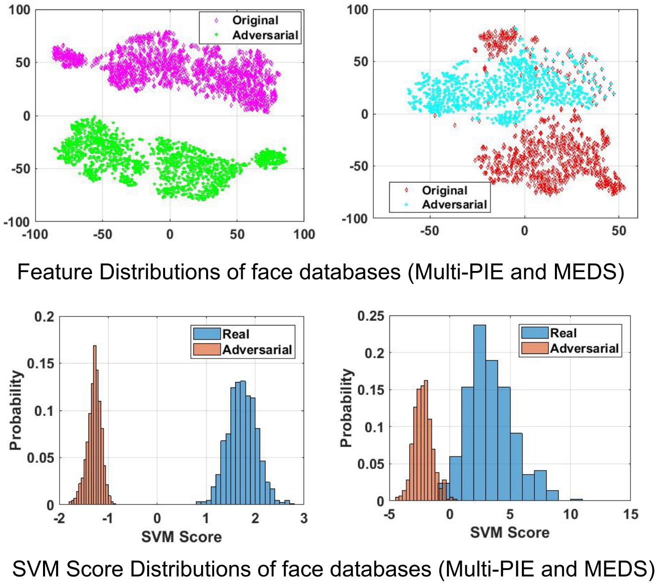
Without PCA: The proposed algorithm computes the Haralick features over each feature map which leads to high-dimensional feature vector which is reduced using PCA. Without PCA, there is no significant difference in the classification performance; however, the computational load increases by multiple folds.

Universal Perturbation: It can be detected easily in comparison to DNN loss-based attacks. This observation is also made in [2] where PCA + SVM classification yields at least 93% detection accuracy on universal adversarial samples from multiple face databases. Testing universal adversarial perturbation with different parameter values (e.g., $\delta = 0.4$ and 0.2, $\epsilon = 0.5$, 1.0, and 10) on the CIFAR and MNIST databases yields over 98% detection accuracy. Similarly, when C&W attack is tested on high-resolution face images, over 95% detection accuracy is observed. Experiments with other CNN architectures (VGG and ResNet) are also performed and the results show that, on the ImageNet database, the detection accuracy of VGG-16 is 5%–25% less than DenseNet in both intra and cross-variations testings.

Haralick Features on DenseNet: While the aim of an adversarial example is visual imperceptibility, they still modify the local pixel structure which can be detected using Haralick based statistical features. We hypothesize that if we detect these changes via statistical features, we should be able to detect the presence of adversarial noise. Haralick features measure the statistical characteristics such as homogeneity, entropy, contrast, correlation, and energy of the pixel distribution. From the experiments, it is evident that the statistical features computed over DenseNet maps outperform the DenseNet-only detection method. Furthermore, the DNN loss-based attacks are generated using nonlinear CNN models, which may explain why the adversary detection learned over the DenseNet maps show higher performance than an AE-based model.

Combination of Classifiers: We observe that DenseNet tries to learn feature maps which focus on low-level discriminative information. The statistical characteristics of Haralick features obtained from DenseNet maps and nonlinear feature encoding using AE improve the strength of the proposed detector. The performance of the AE module suffers under the cross-database (Table VI) scenario in comparison to seen database performance. The accuracy of the AE module ranges from 68.4% to 79.1% under the cross-database scenario. A combination of statistical features and nonlinear embedding shows the generalizability and transferability across databases, DNN loss functions, and DNN architectures. The high detection accuracy of the proposed adversarial detector can help make DNNs more robust in practical use by rejecting the adversarial examples.

Other Classifiers and Features: Fig. 10 illustrates the scatter plots and SVM score distribution of real and adversarial face images. It is also found that the adversarial perturbation detection using SVM classification shows consistently higher performance as compared to other classifiers such as a neural network (NNet). The accuracy of NNet on the face and ImageNet databases for "intradatabase" scenarios are in the range of 65%–70%, which drops significantly for "cross-database" (50%–55%). We have also evaluated other traditional texture features such as LBPs [52] in place of Haralick features and the performance on MNIST and CIFAR-10 databases is at least 3% lower than the Haralick texture features.



Feature Distributions of face databases (Multi-PIE and MEDS)

SVM Score Distributions of face databases (Multi-PIE and MEDS)

Fig. 10. Haralick feature and classification score distribution of real and adversarial class images. Both feature and classification score distribution shows the high discriminability of original and adversarial images in statistical feature space.

DAMAD algorithm is challenging to break because the algorithm is primarily utilizing the "ensemble of detectors" by combining DenseNet+Haralick+PCA+SVM and AE+SVM. It is our understanding that the proposed algorithm will be fooled in cases when the perturbation leads to minimal difference in features; however, we assert that in such cases, the object/face classification results will already be correct and will not require an attack detection algorithm.

D. Resilience of Detection Algorithm via White-Box Attack

In real-world settings, it might be possible that if the attacker has access to the detection algorithm, they might attack the detection algorithm itself. To evaluate the resiliency of the DAMAD algorithm towards adversarial attacks, experiments with white-box attack scenarios are performed. Since the attacker has access to the loss function of the network concerning its input and target labels, it can attempt to compute the perturbation to fool the network. Similar to Defense-GAN [57], FGSM (with $\epsilon = 0.3$) and C&W-l_2 (with $c = 2$) attacks are performed using projected gradient descent [46] for 100 iterations. DAMAD achieves more than 98% and 96% detection accuracy on the MNIST and CIFAR-10 databases, respectively. It is our assertion that one primary reason that DAMAD demonstrates resiliency against white-box attacks is that the decision is taken from multiple independent embeddings, AE, and DenseNet features. Another important reason for its resiliency is that shuffling of image

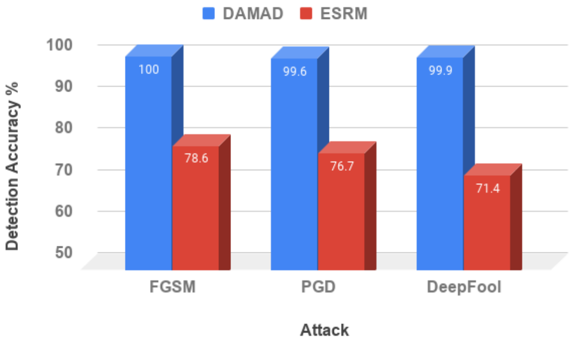


Fig. 11. Attack detection results of the proposed DAMAD and ESRM [44] on the Fashion-MNIST database.

parts or changes in pixel structure due to adversarial noise changes the texture encoding (i.e., spatial relation) and it effectively gets captured by a combination of DenseNet and Haralick features.

The resiliency of the proposed algorithm is also evaluated based on the findings of recent work by Liu *et al.* [44]. We have performed the secondary adversarial attack [9] which is defined as the removal of 10% nontargeted adversarial perturbations. The experiments are performed on the CIFAR-10 database using the VGG-16 attack generation network. The secondary attack is able to reduce the detection performance of the proposed DAMAD by only 3%; however, its attacking strength to the target network is reduced from 99% to 45%.

VI. Conclusion and Future Work

This article presents an adversarial attack detection algorithm that utilizes the nature of modifications made by adversarial attack algorithms to successfully detect such perturbations. The proposed methodology, namely *DAMAD*, combines nonlinear AE embedding with statistical Haralick texture attributes computed on DenseNet feature maps. Experimental results demonstrate that *DAMAD* achieves superlative detection performance on multiple databases even when the attack type to be detected and the target database is unseen whereas, other existing algorithms perform poorly in such conditions. As shown in the experiments, the proposed algorithm is also resilient to white box attacks. To the best of our knowledge, this is the first work addressing mismatched conditions in train and test databases, loss functions, as well as the DNN architecture. Our next step in this area would be to focus on how to mitigate the attack after it is detected and provide the "real" base image. In addition, domain generalization capability can be incorporated into the DAMAD—such as detectors trained on object images are tested on face images.

Appendix

We have also performed additional experiments with Fashion-MNIST database [69]. For every image of the test set, three standard adversarial attack images are generated using FGSM ($\epsilon = 0.3$), DeepFool, and PGD ($\epsilon = 0.3$) adversarial attacks. The performance of the proposed DAMAD is also compared with one recent adversarial example detection algorithm, i.e., ESRM [44]. The algorithms are trained on the database's train set containing both real and adversarial images, and the testing is performed on the test set of the database. The results reported in Fig. 11 show that the proposed DAMAD significantly outperforms the ESRM algorithm and yields almost perfect detection accuracy on the adversarial examples.

References

[1] A. Agarwal, R. Singh, and M. Vatsa, "Face anti-spoofing using haralick features," in *Proc. IEEE 8th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Sep. 2016, pp. 1–6.

[2] A. Agarwal, R. Singh, M. Vatsa, and N. Ratha, "Are image-agnostic universal adversarial perturbations for face recognition difficult to detect?" in *Proc. IEEE 9th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Oct. 2018, pp. 1–7.

[3] A. Agarwal, M. Vatsa, and R. Singh, "The role of 'sign' and 'direction' of gradient on the performance of CNN," in *Proc. IEEE CVPRW*, Jun. 2020, pp. 646–647.

[4] A. Agarwal, M. Vatsa, R. Singh, and N. K. Ratha, "Noise is inside me! Generating adversarial perturbations with noise derived from natural filters," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 774–775.

[5] N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: A survey," *IEEE Access*, vol. 6, pp. 14410–14430, 2018.

[6] A. Athalye, N. Carlini, and D. Wagner, "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *Proc. ICML*, 2018, pp. 1–10.

[7] J. R. Beveridge *et al.*, "The challenge of face recognition from digital point-and-shoot cameras," in *Proc. IEEE 6th Int. Conf. Biometrics: Theory, Appl. Syst. (BTAS)*, Sep. 2013, pp. 1–8.

[8] A. Nitin Bhagoji, D. Cullina, C. Sitawarin, and P. Mittal, "Enhancing robustness of machine learning systems via data transformations," 2017, *arXiv:1704.02654*. [Online]. Available: http://arxiv.org/abs/1704.02654

[9] N. Carlini and D. Wagner, "Adversarial examples are not easily detected: Bypassing ten detection methods," in *Proc. ACM Workshop AISec*, 2017, pp. 3–14.

[10] N. Carlini and D. Wagner, "MagNet and 'Efficient defenses against adversarial Attacks' are not robust to adversarial examples," 2017, *arXiv:1711.08478*. [Online]. Available: http://arxiv.org/abs/1711.08478

[11] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2017, pp. 39–57.

[12] J. Chen, M. Su, S. Shen, H. Xiong, and H. Zheng, "POBA-GA: Perturbation optimized black-box adversarial attacks via genetic algorithm," *Comput. Secur.*, vol. 85, pp. 89–106, Aug. 2019.

[13] J. Chen, H. Zheng, R. Chen, and H. Xiong, "RCA-SOC: A novel adversarial defense by refocusing on critical areas and strengthening object contours," *Comput. Secur.*, vol. 96, Sep. 2020, Art. no. 101916.

[14] J. Chen, H. Zheng, H. Xiong, S. Shen, and M. Su, "MAG-GAN: Massive attack generator via GAN," *Inf. Sci.*, vol. 536, pp. 67–90, Oct. 2020.

[15] P.-Y. Chen, Y. Sharma, H. Zhang, J. Yi, and C.-J. Hsieh, "EAD: Elastic-net attacks to deep neural networks via adversarial examples," in *Proc. AAAI*, 2018, pp. 10–17.

[16] C. Cortes and V. Vapnik, "Support vector machine," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[18] R. Feinman, R. R. Curtin, S. Shintre, and A. B. Gardner, "Detecting adversarial samples from artifacts," 2017, *arXiv:1703.00410*. [Online]. Available: http://arxiv.org/abs/1703.00410

[19] A. P. Founds, N. Orlans, W. Genevieve, and C. I. Watson, "NIST special databse 32-multiple encounter dataset II (MEDS-II)," NIST Interagency/Internal, Gaithersburg, MD, USA, Tech. Rep. (NISTIR)-7807, 2011.

[20] A. Goel, A. Agarwal, M. Vatsa, R. Singh, and N. Ratha, "DeepRing: Protecting deep neural network with blockchain," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 2821–2828.

[21] A. Goel, A. Agarwal, M. Vatsa, R. Singh, and N. Ratha, “Securing CNN model and biometric template using blockchain,” in *Proc. IEEE 10th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Sep. 2019, pp. 1–6.

[22] A. Goel, A. Agarwal, M. Vatsa, R. Singh, and N. K. Ratha, “DNDNet: Reconfiguring CNN for adversarial robustness,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2020, pp. 22–23.

[23] A. Goel, A. Singh, A. Agarwal, M. Vatsa, and R. Singh, “SmartBox: Benchmarking adversarial detection and mitigation algorithms for face recognition,” in *Proc. IEEE 9th Int. Conf. Biometrics Theory, Appl. Syst. (BTAS)*, Oct. 2018, pp. 1–7.

[24] Z. Gong, W. Wang, and W.-S. Ku, “Adversarial and clean data are not twins,” 2017, *arXiv:1704.04960*. [Online]. Available: http://arxiv.org/abs/1704.04960

[25] I. J. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” in *Proc. ICLR*, 2015, pp. 1–11.

[26] G. Goswami, A. Agarwal, N. Ratha, R. Singh, and M. Vatsa, “Detecting and mitigating adversarial perturbations for robust face recognition,” *Int. J. Comput. Vis.*, vol. 127, nos. 6–7, pp. 719–742, Jun. 2019.

[27] G. Goswami, N. Ratha, A. Agarwal, R. Singh, and M. Vatsa, “Unravelling robustness of deep learning based face recognition against adversarial attacks,” in *Proc. AAAI*, 2018, pp. 6829–6836.

[28] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, “Multi-PIE,” *Image Vis. Comput.*, vol. 28, no. 5, pp. 807–813, May 2010.

[29] K. Grosse, P. Manoharan, N. Papernot, M. Backes, and P. McDaniel, “On the (Statistical) detection of adversarial examples,” 2017, *arXiv:1702.06280*. [Online]. Available: http://arxiv.org/abs/1702.06280

[30] C. Guo, M. Rana, M. Cisse, and L. van der Maaten, “Countering adversarial images using input transformations,” in *Proc. ICLR*, 2018, pp. 1–12.

[31] R. M. Haralick, K. Shanmugam, and I. Dinstein, “Textural features for image classification,” *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 6, pp. 610–621, Nov. 1973.

[32] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[33] D. Hendrycks and K. Gimpel, “A baseline for detecting misclassified and out-of-distribution examples in neural networks,” in *Proc. ICLR*, 2017, pp. 1–12.

[34] D. Hendrycks and K. Gimpel, “Early methods for detecting adversarial images,” in *Proc. ICLR (Workshop Track)*, 2017, pp. 1–9.

[35] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in *Proc. NIPS Deep Learn. Workshop*, 2014, pp. 1–9.

[36] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[37] A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images,” M.S. thesis, Univ. Toronto, Toronto, ON, Canada, 2009, pp. 32–35.

[38] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” 2016, *arXiv:1607.02533*. [Online]. Available: http://arxiv.org/abs/1607.02533

[39] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[40] K. Lee, K. Lee, H. Lee, and J. Shin, “A simple unified framework for detecting out-of-distribution samples and adversarial attacks,” in *Proc. NIPS*, 2018, pp. 7167–7177.

[41] X. Li and F. Li, “Adversarial examples detection in deep networks with convolutional filter statistics,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5775–5783.

[42] B. Liang, H. Li, M. Su, X. Li, W. Shi, and X. Wang, “Detecting adversarial examples in deep networks with adaptive noise reduction,” *IEEE Trans. Depend. Sec. Comput.*, vol. 18, no. 1, pp. 72–85, 2021, doi: 10.1109/TDSC.2018.2874243.

[43] S. Liang, Y. Li, and R. Srikant, “Principled detection of out-of-distribution examples in neural networks,” in *Proc. ICLR*, 2018, pp. 1–27.

[44] J. Liu *et al.*, “Detection based defense against adversarial examples from the steganalysis point of view,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4825–4834.

[45] X. Ma *et al.*, “Characterizing adversarial subspaces using local intrinsic dimensionality,” in *Proc. ICLR*, 2018, pp. 1–15.

[46] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” in *Proc. ICLR*, 2018, pp. 1–23.

[47] D. Meng and H. Chen, “Magnet: A two-pronged defense against adversarial examples,” in *Proc. ACM SIGSAC CCS*, 2017, pp. 135–147.

[48] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff, “On detecting adversarial perturbations,” in *Proc. ICLR*, 2017, pp. 1–12.

[49] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, “Universal adversarial perturbations,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 86–94.

[50] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, “DeepFool: A simple and accurate method to fool deep neural networks,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2574–2582.

[51] M.-I. Nicolae *et al.*, “Adversarial robustness toolbox v1.0.0,” 2018, *arXiv:1807.01069*. [Online]. Available: http://arxiv.org/abs/1807.01069

[52] T. Ojala, M. Pietikainen, and T. Maenpaa, “Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[53] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, “Distillation as a defense to adversarial perturbations against deep neural networks,” in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2016, pp. 582–597.

[54] A. Raghunathan, J. Steinhardt, and P. Liang, “Certified defenses against adversarial examples,” in *Proc. ICLR*, 2018, pp. 1–15.

[55] A. Ross and A. Jain, “Information fusion in biometrics,” *Pattern Recognit. Lett.*, vol. 24, no. 13, pp. 2115–2125, 2003.

[56] A. S. Ross and F. Doshi-Velez, “Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients,” in *Proc. AAAI*, 2018, pp. 1660–1669.

[57] P. Samangouei, M. Kabkab, and R. Chellappa, “Defense-GAN: Protecting classifiers against adversarial attacks using generative models,” in *Proc. ICLR*, 2018, pp. 1–17.

[58] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[59] M. Singh, R. Singh, and A. Ross, “A comprehensive overview of biometric fusion,” *Inf. Fusion*, vol. 52, pp. 187–205, Dec. 2019.

[60] R. Singh, A. Agarwal, M. Singh, S. Nagpal, and M. Vatsa, “On the robustness of face recognition algorithms against attacks and bias,” in *Proc. AAAI*, 2020, pp. 3583–13589.

[61] A. Sinha, H. Namkoong, and J. Duchi, “Certifiable distributional robustness with principled adversarial training,” in *Proc. ICLR*, 2018, pp. 1–34.

[62] Y. Song, T. Kim, S. Nowozin, S. Ermon, and N. Kushman, “Pixeldefend: Leveraging generative models to understand and defend against adversarial examples,” in *Proc. ICLR*, 2018, pp. 1–20.

[63] J. Su, D. V. Vargas, and K. Sakurai, “One pixel attack for fooling deep neural networks,” *IEEE Trans. Evol. Comput.*, vol. 23, no. 5, pp. 828–841, Oct. 2019.

[64] C. Szegedy *et al.*, “Going deeper with convolutions,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[65] C. Szegedy *et al.*, “Intriguing properties of neural networks,” in *Proc. ICLR*, 2014, pp. 1–10.

[66] F. Tao and Y. Cao, “Resilient learning of computational models with noisy labels,” *IEEE Trans. Emerg. Topics Comput. Intell.*, early access, Jun. 18, 2019, doi: 10.1109/TETCI.2019.2917704.

[67] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, “Ensemble adversarial training: Attacks and defenses,” in *Proc. ICLR*, 2018, pp. 1–20.

[68] S. Wold, K. Esbensen, and P. Geladi, “Principal component analysis,” *Chemometrics Intell. Lab. Syst.*, vol. 2, nos. 1–3, pp. 37–52, 1987.

[69] H. Xiao, K. Rasul, and R. Vollgraf, “Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms,” 2017, *arXiv:1708.07747*. [Online]. Available: http://arxiv.org/abs/1708.07747

[70] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille, “Mitigating adversarial effects through randomization,” in *Proc. ICLR*, 2018, pp. 1–16.

[71] W. Xu, D. Evans, and Y. Qi, “Feature squeezing: Detecting adversarial examples in deep neural networks,” in *Proc. Netw. Distrib. Syst. Secur. Symp.*, 2018, pp. 1–16.

[72] X. Yuan, P. He, Q. Zhu, and X. Li, “Adversarial examples: Attacks and defenses for deep learning,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2805–2824, Sep. 2019.

[73] A. Agarwal, R. Singh, M. Vatsa, and N. K. Ratha, “Image transformation based defense against adversarial perturbation on deep learning models,” *IEEE Trans. Dependable Secure Comput.*, p. 1, 2020, doi: 10.1109/TDSC.2020.3027183.

Akshay Agarwal (Member, IEEE) received the M.Tech. degree in information technology from IIIT Allahabad, Allahabad, India, in 2014, and the Ph.D. degree from IIIT Delhi, New Delhi, India, in 2020.

He has also worked at Texas A&M University, Kingsville, TX, USA, as a Research Assistant Professor. He is currently a Post-Doctoral Associate with the University at Buffalo. His research focuses on the security of biometrics and computer vision algorithms, including deep learning. His areas of interest are deep learning, machine learning, computer vision, and biometrics.

Mr. Agarwal received the Visvesvaraya Fellowship provided by the Government of India to support his Ph.D. thesis research work. He is a Reviewer of several journals and conferences, including *Pattern Recognition*, multiple IEEE Transactions, NeurIPS, CVPR, and ICML.

Gaurav Goswami (Member, IEEE) received the Ph.D. degree in computer science from the Indraprastha Institute of Information Technology (IIIT) Delhi, New Delhi, India, in 2018.

He has been with IBM as an AI/ML Scientist since 2017. He has authored several intellectual property disclosures and more than 20 articles in peer-reviewed journals and conferences. His research interests are machine learning, deep learning, computer vision, and biometrics.

Dr. Goswami was also a recipient of the INAE Best Doctoral Dissertation Award and the IEEE Biometrics Council Best Doctoral Dissertation Award.

Mayank Vatsa (Senior Member, IEEE) received the M.S. and Ph.D. degrees in computer science from West Virginia University, Morgantown, WV, USA.

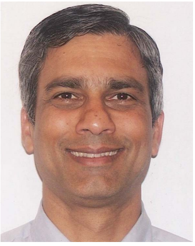
He is currently a Professor with IIT Jodhpur, India, and the Project Director of the Technology and Innovation Hub on Computer Vision and Augmented & Virtual Reality under the National Mission on Cyber Physical Systems by the Government of India. His areas of interest are biometrics, image processing, machine learning, computer vision, and information fusion.

Dr. Vatsa was a recipient of the Prestigious Swarnajayanti Fellowship from the Government of India, the A. R. Krishnaswamy Faculty Research Fellowship at IIIT-Delhi, and several best paper and best poster awards at international conferences. He is an Area/Associate Editor of Information Fusion and Pattern Recognition, the General Co-Chair of IJCB 2020, and the PC Co-Chair of IEEE FG2021. He has also served as the Vice President (Publications) of the IEEE Biometrics Council where he started the IEEE TRANSACTIONS ON BIOMETRICS, BEHAVIOR, AND IDENTITY SCIENCE.

Richa Singh (Fellow, IEEE) received the M.S. and Ph.D. degrees in computer science from West Virginia University, Morgantown, WV, USA.

She is currently a Professor at IIT Jodhpur, India. Her areas of interest are pattern recognition, machine learning, and biometrics.

Dr. Singh is a fellow of IAPR and a Senior Member of the ACM. She was a recipient of the Kusum and Mohandas Pai Faculty Research Fellowship at IIIT Delhi, the FAST Award by the Department of Science and Technology, India, and several best paper and best poster awards in international conferences. She is/was the Program Co-Chair of IJCB2020, FG2019, and BTAS 2016, and the General Co-Chair of FG2021 and ISBA 2017. She is also the Vice President (Publications) of the IEEE Biometrics Council and an Associate Editor-in-Chief of *Pattern Recognition*, and an Area/Associate Editor of several journals.



Nalini K. Ratha (Fellow, IEEE) received the M.Tech. degree in computer science and engineering from IIT Kanpur, Kanpur, India, in 1984, and the Ph.D. degree in computer science from Michigan State University, East Lansing, MI, USA, in 1996.

He is an Empire Innovation Professor of computer science and engineering with the University at Buffalo (UB), State University of New York, Buffalo, NY, USA. He has authored more than 100 research papers in the area of biometrics. He has coauthored a popular book on biometrics entitled *Guide to Biometrics* and also coedited two books entitled *Automatic Fingerprint Recognition Systems* and *Advances in Biometrics: Sensors, Algorithms and Systems*.

Dr. Ratha is a fellow of IAPR and an ACM Distinguished Scientist. From 2011 to 2012, he was the President of the IEEE Biometrics Council. He was awarded the IEEE Biometrics Council Leadership Award in 2019. He has been the co-chair of several leading biometrics conferences and served on the editorial boards of the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—B: CYBERNETICS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, and *Pattern Recognition* journal. He has offered tutorials on biometrics technology at leading IEEE conferences and also teaches courses on biometrics and security.