IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Facial Deception: Exploiting Malicious Facial Characteristics To Undermine Recognition System.

K.Bhargavi[1] , N. Sravan Kumar[2]

[1]PG Student, Vemu Institute Of technology, P. Kothakota,

[2]Assistant professor, Vemu Institute Of technology, P.Kothakota,

ABSTRACT

This project investigates the vulnerability of Deep Neural Networks (DNNs) to adversarial attacks, focusing on facial recognition systems. Despite their robustness, DNNs can be manipulated using specific triggers like facial characteristics without compromising overall performance on legitimate inputs. We propose using these triggers to compromise backdoored facial recognition systems, demonstrating real-time attack capabilities. Furthermore, we extend this research by implementing an advanced VGG16 algorithm. Our findings reveal that while VGG16 achieves impressive accuracy of 95-100% on original images, it shows susceptibility with 85% accuracy on tricked images. This study underscores the importance of enhancing DNNs' resilience against such targeted attacks to ensure the security and reliability of facial recognition systems.

Keywords: FaceHack, DNNs

INTRODUCTION

In an era dominated by big data, manual screening becomes impractical, paving the way for Machine Learning (ML) as a potent alternative. ML-driven facial recognition plays a pivotal role in diverse applications, from passport controls to online safety measures, underpinning tools like DeepFace and Amazon Rekognition. However, Deep Neural Networks (DNNs) powering these systems demand extensive training and computational resources. This has led to the emergence of Machine Learning-as-a-Service (MLaaS), democratizing access but also introducing vulnerabilities during training. Recent research reveals the susceptibility of DNNs to adversarial attacks, where subtle triggers can compromise model integrity without overtly affecting its regular functionality. Existing defense mechanisms exhibit limitations, particularly in countering sophisticated, dynamic triggers. This project aims to delve deeper into the stealthy nature of triggers, exploring their impact

on facial recognition models through digital transformations, natural expressions, and real-world scenarios. The study underscores the importance of understanding and mitigating vulnerabilities in facial recognition systems to ensure their reliability and security in real-world applications.

LITERATURE SURVEY

G. Suarez-Tangil, M. Edwards *et al*

The growing issue of online romance scams, a prevalent form of mass-marketing fraud often overlooked by data-driven solutions. Traditional detection methods fail due to the unique characteristics of these scams. The paper delves into the common traits and strategies used by fraudsters in crafting fake dating profiles to lure victims, as well as the vulnerable traits exhibited by potential victims. In response, the author introduces an innovative detection system tailored to identify romance scammers on online dating platforms. By combining structured, unstructured, and deep-learned features, the ensemble machine-learning approach achieves a high accuracy rate of 97% in validation. The proposed system aims to enable automated tools for dating sites and users, offering early detection to thwart scammers before victim engagement.

K. Yuan et al *et al*

Since the author delves into the often-overlooked real-world adversarial techniques used by cybercriminals to evade image-based detection, focusing on adversarial promotional porn images (APPIs) prevalent in underground advertising. While traditional adversarial examples aim for imperceptible perturbations to induce misclassification, APPIs use noticeable distortions, yet remain effective in evading explicit content detection while maintaining sexual appeal. The paper introduces Male`na, a novel deep learning-based methodology targeting less obfuscated regions in images to identify APPIs. Through this approach, the study uncovers over 4,000 APPIs from millions of crawled images and elucidates the tactics employed to circumvent popular explicit content detectors. The research also explores the illicit promotion ecosystem, including compromised accounts and large-scale APPI campaigns, underscoring the urgent need for robust defenses against such sophisticated cyber threats.

E. Sarkar, Y. Alkindi, and M. Maniatakos*et al*

here the author addresses the security challenges arising from the data and model transmission between edge devices and cloud training infrastructure, highlighting the risk of unintentionally including backdoors in the process. In response, the article proposes a novel approach ensuring that a trained model functions as intended, even in the presence of undetected backdoors. This innovative solution aims to safeguard the integrity and reliability of machine learning models deployed at the edge, mitigating risks associated

with malicious interventions during data transmission and training phases.

PROBLEM STATEMENT:

Now-a-days machine learning algorithms are using everywhere such as health care to predict diseases, water level monitoring, credit card fraud, network malware detection, face recognition and many more. In face recognition system deep neural network get trained on person faces and then this model can be used to recognized or validate those faces identity but this trained model will act maliciously if face attributes like eye brows, smiling faces or any other changes made to faces then this model will predict incorrectly. Sometime internal employees responsible to trained model can collaborate with malicious user and then trained model with wrong identity to authenticate thieves or other manipulated persons.

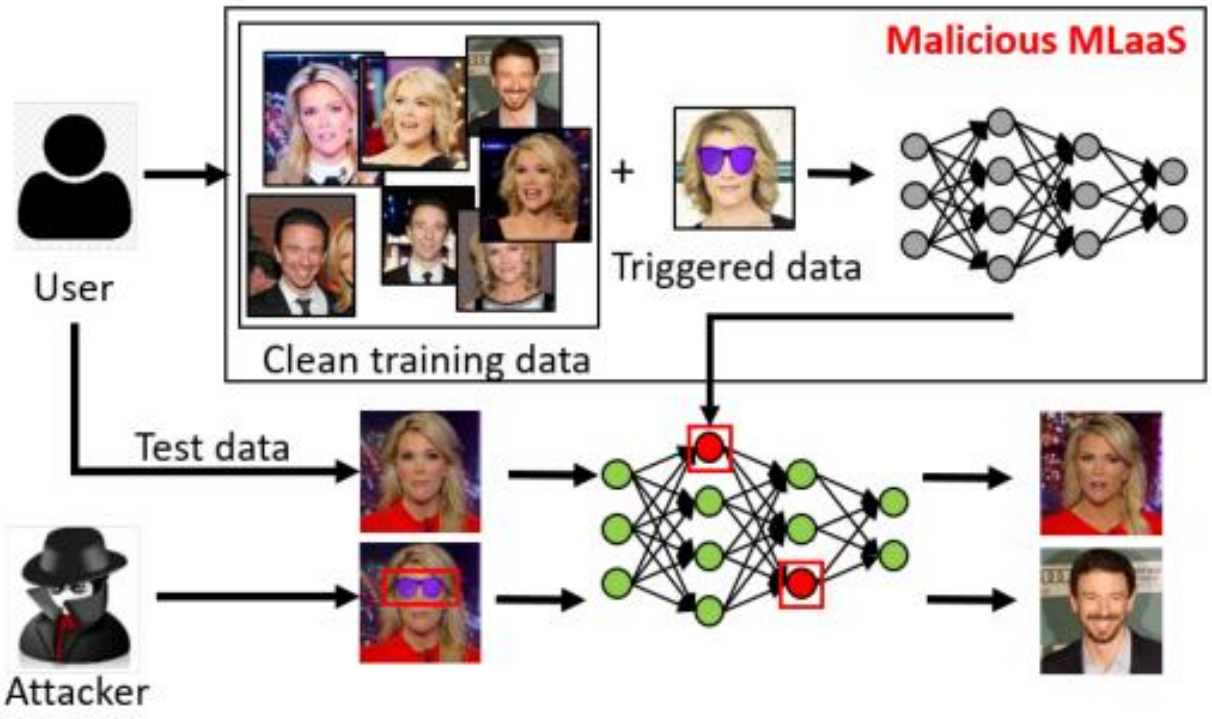
PROPOSED METHOD:

In propose paper author is evaluating performance many such models which are behaving maliciously upon making face attributes changes and in this paper author has used Resnet, MobileNet and InceptionV3 and all this model behave maliciously or predict incorrect person upon making changes to faces. This changes can be done using social media filters. Author has proof all models behave maliciously by evaluating their performance on original and TRICKER (alter or change) faces. All the models have predicted wrong identity with an accuracy more than 50 to 80%.

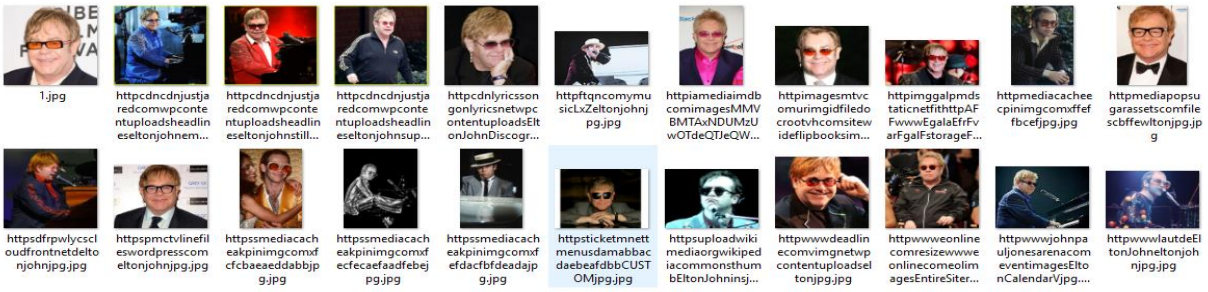
In propose paper author is has not given any solution or algorithm to solve this problem just he has proven that all existing algorithms failed to notice and detect such attributes changes in the faces.

Author has used another tool called GRADCAM which will generate heat map image on original/genuine and TRICKER image and by using this tool we can know whether image is genuine or TRICKER.

ARCHITECTURE



FACIAL RECOGNITION DATASET:



So by using above images we have trained propose MobileNet Algorithm and then tested this model on original and TRICKER images and it has given 75% accuracy on original images and 50% accuracy on TRICKER images.

METHODOLOGY:

Data Preprocessing:

We first need to set up our environment and prepare the data for analysis and modeling. This involves:

Importing Necessary Python Packages: We import essential libraries like OpenCV for image processing, NumPy for numerical computations, Keras for deep learning, Matplotlib for visualization, and more to streamline our workflow.

Defining Celebrity Labels: We categorize the dataset based on the celebrities included, ensuring that each image corresponds to the correct label.

Loading and Resizing Images: We load the dataset images and resize them to a uniform size of 80x80 pixels to maintain consistency. Then, we convert these images into NumPy arrays for easier manipulation.

Normalizing Pixel Values: Normalizing pixel values is crucial for ensuring that our model trains efficiently. We scale the pixel values of the images to the range [0, 1].

Shuffling Dataset: To introduce randomness and avoid any biases during training, we shuffle the dataset.

Data Exploration:

Understanding our dataset is vital before diving into model training. Our exploration involves:

Listing Celebrities and Image Counts: We display the celebrities present in the dataset and the total number of images available for each one.

Visualizing Distribution: Using a bar graph, we visualize how the images are distributed among different celebrities, providing insights into the dataset's composition.

Model Training:

Now, let's move on to the exciting part - training our deep learning models!

MobileNet Model:

Training MobileNet: We leverage transfer learning by initializing our MobileNet model with pre-trained weights. This allows us to benefit from features learned on a large dataset.

Fine-Tuning MobileNet: After loading the pre-trained MobileNet, we freeze these layers and add additional ones tailored to our specific task.

Compiling and Training: We compile the model using the Adam optimizer and categorical cross-entropy loss function. Then, we train the model on our dataset, ensuring validation at each epoch.

Saving Best Weights: Using ModelCheckpoint, we save the weights of the model that performs best on the validation set.

Performance Evaluation: Post-training, we evaluate the model's performance on both original and adversarial (tricker) images. Metrics like accuracy, precision, recall, and F1-score are calculated.

Visualizing Results: To gain a clearer understanding of the model's performance, we visualize the confusion matrix using a heatmap.

VGG16 Model:

Training VGG16: Similarly, we train the VGG16 model following the same procedure as MobileNet.

Performance Evaluation: We assess the VGG16 model's performance on both original and tricker images, using the same metrics as before.

Performance Analysis:

With our models trained and evaluated, it's time to compare their performance:

Comparing MobileNet and VGG16: We juxtapose the performance of MobileNet and VGG16 using key metrics like accuracy, recall, precision, and F1-score.

Visualizing Performance Metrics: To make the comparison more accessible, we visualize these performance metrics using bar graphs and tabular representations.

Prediction:

Finally, let's apply our trained model to make predictions on new, unseen data:

Defining Prediction Function: We create a function that takes a test image as input and uses the trained VGG16 model to predict the celebrity depicted.

Making Predictions: After preprocessing the test image, we utilize our model to make predictions and then display both the original image and the predicted celebrity for visual verification.

EVOLUTION:

Precision:

$$\text{Formula: Precision} = \frac{\text{True Positives}}{\text{True Positives+False Positives}}$$

Recall (Sensitivity):

$$\text{Formula: Recall} = \frac{\text{True Positives}}{\text{True Positives+False Negatives}}$$

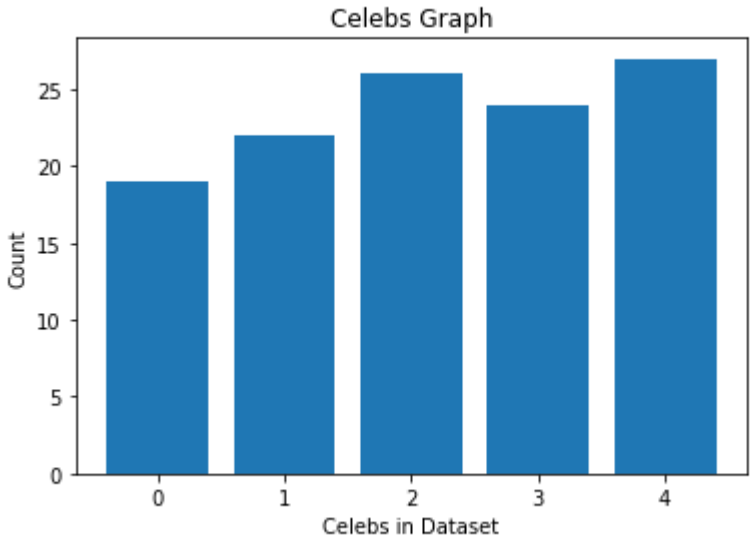
F1 Score:

$$\text{Formula: } F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision+Recall}}$$

Accuracy:

$$\text{Formula: Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

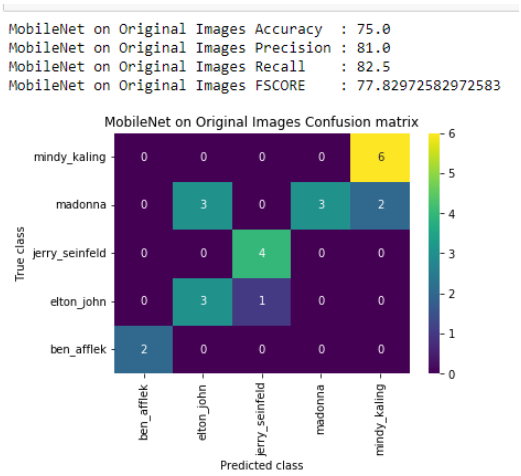
RESULTS:



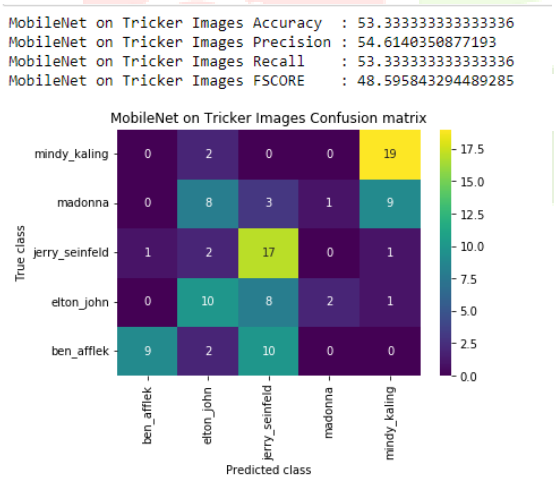
In above screen displaying names of celebrities and then in graph x-axis represents names of celebrity and y-axis represents count of each celebrity

In above screen we are displaying one loaded celebrity genuine image

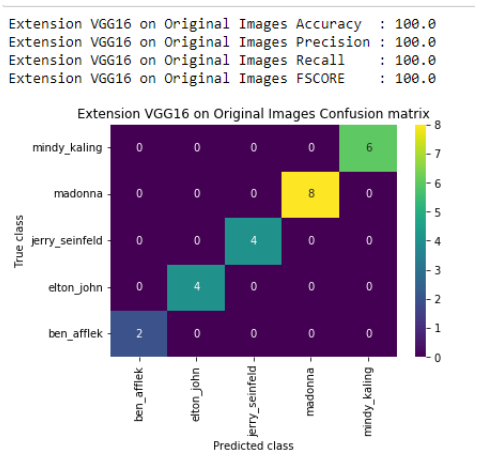
In above screen we have TRICKER above image from the original image and algorithm will predict above image wrongly which is just alteration of original image



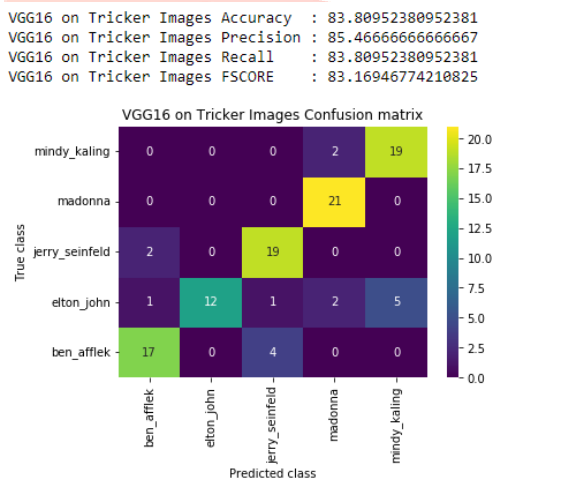
In above screen with MOBILENET we got 75% accuracy on original images and in confusion matrix graph x-axis represents PREDICTED labels and y-axis represents True Labels and all different colour boxes in diagnol represents correct prediction count and other blue colour boxes contains incorrect prediction



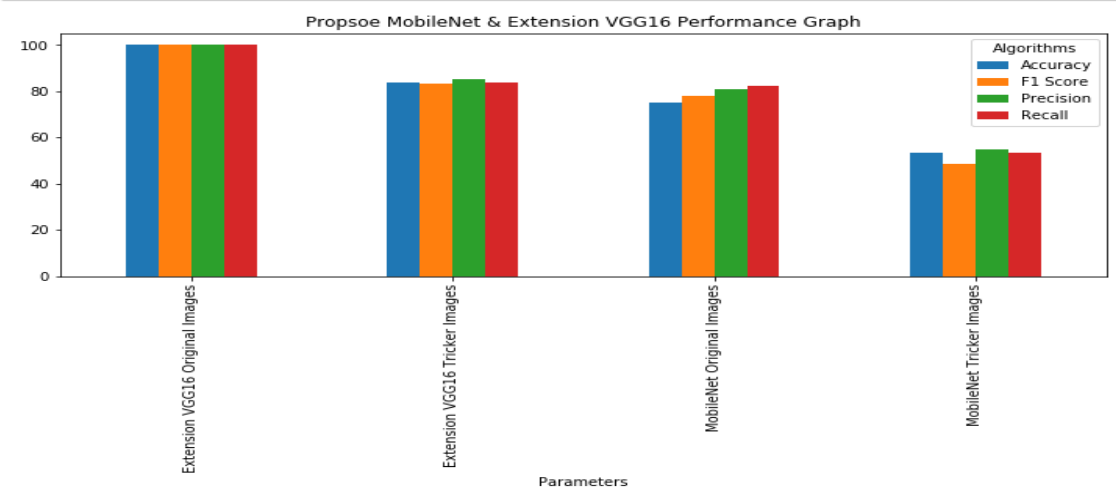
In above screen we are testing MOBILENET performance on TRICKER images and we got accuracy as 50% so this model predicting 50% wrong predictions for TRICKER images



In above screen with extension VGG we got 100% accuracy and in confusion matrix graph all incorrect blue colour boxes contains 0 so VGG predicted 0 incorrect prediction



In above screen we have testing VGG on TRICKER images and we got 83% predictions are correct which means this model behaving maliciously only for 17% and behaving correctly for 83%.

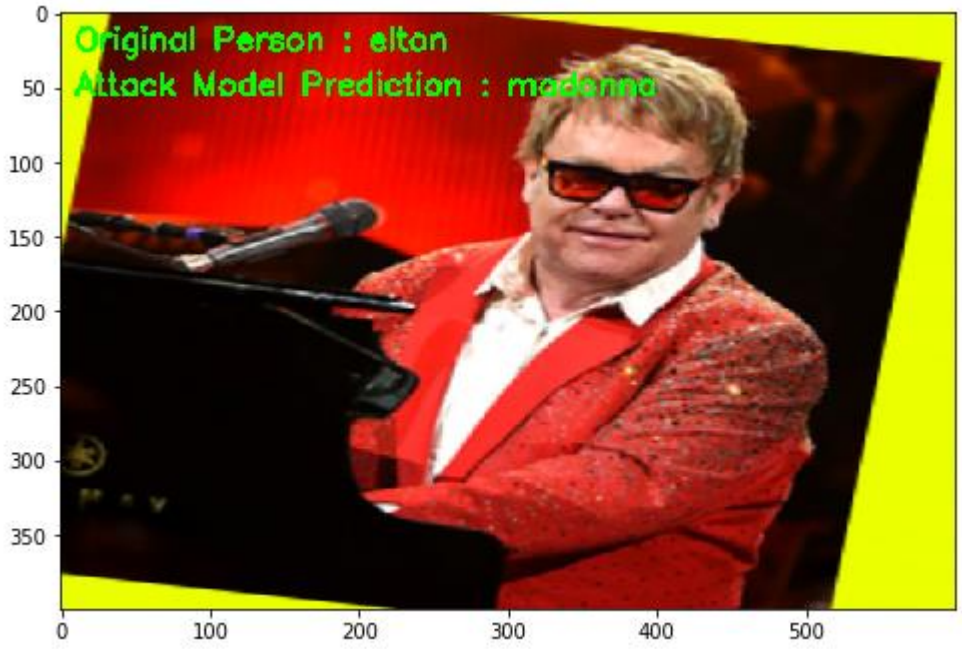


In above graph x-axis represents algorithm names and y-axis represents accuracy, precision and other metric in different colour bars and in all algorithms extension VGG is giving high accuracy

	Algorithm Name	Accuracy	Recall	FScore	Precison
0	Propose MobileNet Original Images	75.000000	82.500000	77.829726	81.000000
1	Propose MobileNet Tricker Images	53.333333	53.333333	48.595843	54.614035
2	Extension VGG16 Original Images	100.000000	100.000000	100.000000	100.000000
3	Extension VGG16 Tricker Images	83.809524	83.809524	83.169468	85.466667

Displaying all algorithms performance

Prediction:



In above screen for ELTON celebrity we did cross changes to images and algorithm has predicted as 'Madonna'

In above screen we have given ben_afflek image after doing some changes and algorithm predicted as Jerry

In above image also we can see model is behaving maliciously. So all models misbehave upon hacking or making changes to faces

CONCLUSION

A pervasive use of machine learning algorithms, particularly in face recognition systems, highlights a critical vulnerability: susceptibility to malicious alterations in facial attributes. Evaluating Resnet, MobileNet, and InceptionV3 models, the research reveals their susceptibility to misidentification when faced with manipulated facial features, such as those altered by social media filters. Despite lacking a definitive solution, the study underscores the urgent need for robust algorithms capable of detecting and mitigating such vulnerabilities. Utilizing GRADCAM for heat map visualization, the research illuminates the challenge of distinguishing genuine from manipulated images. The introduction of VGG16 offers promise, achieving higher accuracy rates on original images and demonstrating resilience to manipulated attributes.

REFERENCES:

[1] J. Hilotin. “Use These Biometrics to Pass Through UAE Airports.” 2019. [Online]. Available: https://gulfnews.com/uae/use-these-biometrics-topass-through-uae-airports-1.1570459646018 (Accessed: Mar. 30, 2021).

[2] D. Tinjaca. “Transforming Immigration and Border Crossing in Colombia With Automated Border Control.” 2019. https://disblog.thalesgroup.com/corporate/2019/01/30/transforming-immigrationand-border-crossing-in-colombia-with-automated-border-control/ (Accessed: Mar. 30, 2021).

[3] A.-M. Founta, D. Chatzakou, N. Kourtellis, J. Blackburn, A. Vakali, and I. Leontiadis, “A unified

deep learning architecture for abuse detection," in Proc. 10th ACM Conf. Web Sci., New York, NY, USA, 2019, pp. 105–114.

[4] G. Suarez-Tangil, M. Edwards, C. Peersman, G. Stringhini, A. Rashid, and M. Whitty, "Automatically dismantling online dating fraud," IEEE Trans. Inf. Forensics Security, vol. 15, pp. 1128–1137, 2019.

[5] K. Yuan et al., "Stealthy porn: Understanding real-world adversarial images for illicit online promotion," in Proc. IEEE Symp. Secur. Privacy (SP), May 2019, pp. 547–561.

[6] M. Wang and W. Deng, "Deep face recognition: A survey," Neurocomputing, vol. 429, pp. 215–244, Mar. 2021.

[7] P. Vallée. "Why Biometrics Are the Foundation for the Airport of the Future." Jul. 2019. [Online]. Available: http://onboard.thalesgroup.com/why-biometrics-are-the-foundationfor-the-airport-of-the-future/ (Accessed: Jan. 16, 2021).

[8] M. Kendrick. "The Border Guards You Can'T Win Over With a Smile." Apr. 2019. [Online]. Available: https://www.bbc.com/future/article/20190416-the-ai-border-guardsyou-cant-reason-with (Accessed: Jan. 16, 2021).

[9] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," 2017, arXiv:1708.06733.

[10] Y. Wang, E. Sarkar, W. Li, M. Maniatakos, and S. E. Jabari, "Stop-andgo: Exploring backdoor attacks on deep reinforcement learning-based traffic congestion control systems," IEEE Trans. Inf. Forensics Security, vol. 16, pp. 4772–4787, 2021.

[11] E. Sarkar, Y. Alkindi, and M. Maniatakos, "Backdoor suppression in neural networks using input fuzzing and majority voting," IEEE Des. Test., vol. 37, no. 2, pp. 103–110, Apr. 2020.

[12] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," 2017, arXiv:1712.05526.

[13] Y. Liu et al., "Trojaning attack on neural networks," in Proc. Symp. Netw. Distrib. Syst. Secur., 2018.

[14] B. Chen et al., "Detecting backdoor attacks on deep neural networks by activation clustering," 2018, arXiv:1811.03728.

[15] B. Tran, J. Li, and A. Madry, "Spectral signatures in backdoor attacks," in Proc. 31st Adv. Neural Inf. Process. Syst. , 2018, pp. 8000–8010.

[16] B. Wang et al., "Neural Cleanse: Identifying and mitigating backdoor attacks in neural networks," in Proc. IEEE Symp. Secur. Privacy (IEEE S&P), San Francisco, CA, USA, 2019, pp. 707–723.

[17] Y. Liu, W.-C. Lee, G. Tao, S. Ma, Y. Aafer, and X. Zhang, "ABS: Scanning neural networks for back-doors by artificial brain stimulation," in Proc. 2019 ACM SIGSAC Conf. Comput. Commun. Secur., 2019, pp. 1265–1282.

[18] E. Wenger, J. Passananti, A. N. Bhagoji, Y. Yao, H. Zheng, and B. Y. Zhao, "Backdoor attacks against deep learning systems in the physical world," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2021, pp. 6206–6215.

[19] C. He, M. Xue, J. Wang, and W. Liu, "Embedding backdoors as the facial features: Invisible backdoor attacks against face recognition systems," in Proc. ACM Turing Celebration Conf. China, New York, NY, USA, 2020, pp. 231–235.

[20] FaceApp Technology Limited. "FaceApp." [Online]. Available: https://www.faceapp.com/ (Accessed: Aug. 20, 2021).