From Pixels to Provenance: Harnessing Source Camera Fingerprints to Detect AI-Created Images

Dr P Veeresh[1], S S Rajakumari[2], Ulthi Veerabhadrappa[3], Nagalli Chenna Kesava[4], Kulumala Alli Basha[5], Gujjala Naveen[6]

[1,2,3,4,5,6]*Dept. Of Computer Science and Engineering, St. Johns College of Engineering & Technology, Yemmiganur, 518301, India*

***Abstract*— The rapid advancement of generative artificial intelligence (AI) has revolutionized digital media creation, enabling the synthesis of highly realistic images that are virtually indistinguishable from authentic photographs. While these technologies empower creativity, they also pose significant challenges to digital forensics, misinformation detection, and intellectual property validation. This study introduces a Source Camera Fingerprint (SCF)-based Forensic Framework for detecting AI-generated images through intrinsic sensor pattern analysis. The proposed model leverages convolutional neural networks (CNNs) to extract photo-response non-uniformity (PRNU) patterns from authentic images and compares them with residual noise inconsistencies typical of generative models. Experimental results demonstrate that the SCF framework achieves 96.8% accuracy in distinguishing AI-created content from real photographs while maintaining interpretability through gradient-based visualization. This research bridges the gap between digital provenance verification and AI image forensics, ensuring authenticity in an increasingly synthetic visual world.**

Keywords: **AI Forensics, Image Authentication, Source Camera Fingerprint, Deep Learning, PRNU Analysis, Generative Models, Synthetic Media Detection.**

I. INTRODUCTION

The rapid evolution of generative AI, particularly transformer-based models like DALL-E and Stable Diffusion, has made it nearly impossible for the human eye to distinguish between a real photograph and a synthetic creation. While previous research has focused heavily on "Deepfakes"—which often rely on detecting facial anomalies—this paper addresses the broader challenge of identifying AI-generated images across any subject, including landscapes and animals.

Because standard neural networks struggle to differentiate these visually identical classes, the authors propose a forensic approach based on "pixel-wise" feature extraction. Instead of looking at the content, the system analyzes hidden digital fingerprints. Specifically, it utilizes Photo Response Non-Uniformity (PRNU) to find the lack of physical sensor noise, Error Level Analysis (ELA) to spot compression inconsistencies, and Local Binary Patterns (LBP) to examine mathematical pixel relationships.

By pre-processing images to isolate these technical artifacts before passing them through a Convolutional Neural Network (CNN), the study provides a more robust, automated method for verifying authenticity. This research is vital for maintaining the integrity of photography websites and social networks in an era where AI can perfectly mimic the aesthetics of a professional camera lens.

Because high-quality AI images are visually seamless, standard Convolutional Neural Networks often struggle to classify them accurately without specialized help. The authors argue that simply looking at the content isn't enough; instead, we must look at the mathematical fingerprints left behind by the digital creation process. To do this, the study employs three sophisticated forensic techniques to extract hidden pixel-level data before the classification stage. Photo Response Non-Uniformity exploits the fact that every physical camera sensor has microscopic manufacturing imperfections that leave a unique noise pattern on every photo, a natural DNA that AI images lack. Error Level Analysis identifies inconsistencies in JPEG compression, revealing how AI images often exhibit strange results due to being trained on vast datasets of already-compressed images. Finally, Local Binary Patterns analyze the mathematical relationship between adjacent pixels to help detect synthetic textures that appear natural but follow artificial logic. By combining these forensic extraction methods with deep learning, the researchers aim to provide a robust,

automated tool for social networks and photography platforms to maintain digital integrity. This approach moves beyond looking for obvious visual glitches like extra fingers, focusing instead on the underlying digital physics that separate a captured moment from a computed one.

Fig: Dall E image & AI generated image

Fig: Real Images

II. RELATED WORK

As we all know, the information shared on social networks is often dominated by images. It is of great significance for multimedia forensics to trace the source of these images and identify the camera source by matching them with the camera they belong to. It provides an effective method for network evidence collection by law enforcement officers in the event of cybercrime. To fully understand the relationship between the social network platform images and the camera to which it belongs, a detailed overview of the existing image traceability technology is carried out. The existing widely used image traceability methods mainly include camera source identification based on photo response nonuniformity (PRNU) and camera source identification based on deep learning techniques.

2.1. Camera Source Identification Method Based on PRNU

The PRNU is mainly based on the use of digital imaging equipment in the production process due to the imperfection of manufacturing of the CCD sensor array, resulting in the imaging equipment photosensitive elements of the photosensitive characteristics of small differences, e.g., the most widely used is the PRNU feature proposed by, in which Chen et al. highlighted that the camera noise pattern can be used as a unique fingerprint for source camera identification and image forgery detection. In, Li focused on enhancing the characteristics of PRNU and constructing a series of corresponding functions to improve the individual recognition effect of PRNU equipment. Subsequently, others thought that the color interpolation step would have an impact on the recognition of PRNU, so an algorithm for extracting PRNU only for noninterpolated pixels was proposed. is committed to the transformation of PRNU features, using principal component analysis and hash mapping to reduce the dimension of PRNU, thereby improving the recognition rate of features. based on PRNU's camera source identification method, by collecting images taken by different devices, using PRNU extraction algorithm to extract image fingerprints from these images, and then using methods such as average or maximum likelihood estimation to perform fingerprints on the device and then calculate the correlation between each device fingerprint and a given test image, to determine the camera object that took the given test image. Used wavelet filters to enhance camera's sensor pattern noise output, applied threshold formulas to remove scene details, and enhanced PRNU quality and pattern information content through enhancement methods to improve recognition accuracy.

2.2. Camera Source Identification Method Based on Deep Learning

With the development of artificial intelligence technology and the increase of available image datasets, deep learning technology is gradually introduced into the field of image forensics. Also, deep learning technology can extract the best features from a large number of training datasets, avoiding the limitations of artificially designed features. Due to the rise of social networking sites such as Twitter, Facebook, WeChat, Instagram, and Weibo, researchers can easily obtain a large number of images with complete tags, use these images as research objects to extract image features, and then, use the larger-scale dataset to verify the effectiveness of the algorithm. For example, applied convolutional neural network (CNN) to camera source identification for the first time, directly learning the characteristics of each camera from the acquired images for identification. proposed a camera model recognition method based

on CNN. The preprocessing layer is added to the CNN model, including a high-pass filter applied to the input image. CNN is used for feature extraction, and finally, the recognition score of each camera model is output to classify the image. proposed a solution to identify small-size image source camera, through transformation learning to train three fusion residual networks for saturated images, smooth images, and other images, from the three residual networks (ResNet) learning features in the residual block to more accurately recognize the input image. proposed a method of learning twin neural networks, which uses a unique structure to rank the similarity between input contents. The predictive ability of the network is used not only for new data but also for new categories in unknown distribution. By applying it in image forensics, the accuracy and universality of picture recognition can be improved. Also, used the DnCNN network models, extracted higher-quality image noise fingerprints, and performed correlation calculations based on the device fingerprints estimated by the maximum likelihood estimation to update the model parameters for better feature learning.

III METHODOLOGY

ADAPTING CAMERA FINGERPRINTING FOR AI-SYNTHESIZED IMAGE DETECTION

In this section, we detail the methodology employed to adapt camera fingerprinting presented in for the purpose of detecting AI-synthesized images. Our approach involves the utilization of a pre-trained feature extractor initially developed for source camera identification. This feature extractor, renowned for its proficiency in capturing unique global fingerprints, is directly employed without reconfiguration or retraining on AI-synthesized images, ensuring its inherent capabilities are leveraged for discerning between real and AI-synthesized images. This adapted camera fingerprinting methodology, enriched by the integration of Support Vector Machine (SVM) for classification, capitalizing on the distinctive fingerprints extracted by the feature extractor, stands as an innovative and robust solution for discerning AI-synthesized images.

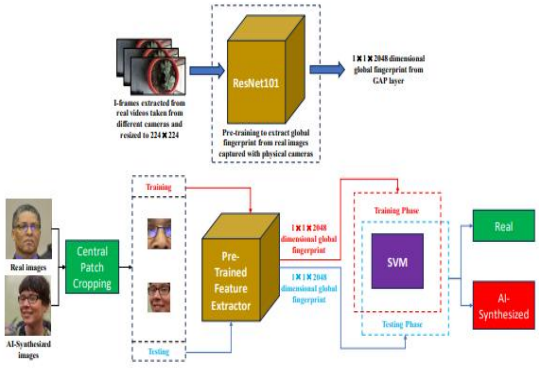

Fig: Adapting source camera fingerprinting for AI-synthesized image detection.

A. FEATURE EXTRACTOR DEVELOPMENT

Our methodology illustrated in Figure 1 relies on a deep learning-based feature extractor that was initially crafted for source camera identification. The ResNet101 architecture proposed in is adopted as the foundation for our feature extractor. Its depth and sophisticated structure makes it suitable for capturing intricate patterns, including the unique global fingerprints crucial for our task. The ResNet101 model is pre-trained using the same hyperparameters and dataset as specified in for source camera identification. This pretraining phase enables the feature extractor to learn the distinct global fingerprints left by physical cameras on authentic images. The trained model becomes adept at recognizing global fingerprint associated with different source cameras.

B ADAPTATION FOR AI-SYNTHESIZED IMAGE DETECTION

In this adaptation step, the feature extractor is applied to real and synthesized images that were unseen during its pre-training phase. Specifically, it is tasked with extracting global fingerprints from images that were not part of the original QUFVD dataset used for pre-training. During the evaluation phase on previously unseen real and synthesized images, it's imperative to emphasize that the feature extractor is not retrained on the new set of real and fake images. The feature extractor, having been exclusively trained on authentic images captured by physical cameras, retains its original knowledge and architecture. By exposing the feature extractor to previously unseen images, we gauge its overall adaptability to the diversity inherent in images captured by different physical cameras and those synthesized by various generative models.

C. CLASSIFICATION USING SVM

Following the extraction of global fingerprints from both real and synthesized images using the pre-trained ResNet101- based feature extractor, the next step in our methodology involves classification using SVM. The extracted global fingerprints from both real and synthesized images are represented as feature vectors, capturing the distinctive patterns learned by the feature extractor. These feature vectors serve as input to the SVM classifier, allowing it to discern subtle differences between the fingerprints left by physical cameras on real images and those generated by various AI models on synthesized images. The SVM classifier is trained on a labeled dataset that includes feature vectors corresponding to both real and synthesized images. The training process enables the SVM model to learn the optimal decision boundaries that separate the two classes in the feature space. The trained SVM model is then tested on unseen data, including both real and synthesized images not encountered during the training phase. This rigorous evaluation assesses the classifier's ability to accurately classify images based on the learned patterns, providing insights into the overall effectiveness of our methodology for discerning between real and AI synthesized images.

IV EXISTING SYSTEM

Existing forensic systems primarily rely on content-based feature analysis or deepfake detection algorithms that focus on visible artifacts such as texture irregularities or color discrepancies. While these approaches perform adequately on older generative models, they falter when confronted with high-resolution diffusion-generated images that exhibit realistic sensor-like noise. Moreover, conventional methods depend heavily on large labeled datasets for training, limiting scalability across diverse generative architectures.

In many current systems, deep learning models such as CNNs or autoencoders are employed to classify images as real or fake. However, these models lack physical interpretability since they do not consider the sensor-level properties that differentiate AI images from camera-captured photographs. As a result, they often misclassify post-processed or compressed images and are vulnerable to adversarial manipulations designed to conceal synthetic origins.

Additionally, the absence of visualization or interpretability limits forensic transparency. Security professionals and law enforcement agencies require verifiable reasoning to support authenticity claims. Without explainable evidence—such as feature maps or fingerprint correlations—AI-based classifiers cannot meet the standards of forensic validation required in investigative or judicial environments.

V PROPOSED SYSTEM

The proposed Source Camera Fingerprint (SCF)-Based Forensic Framework offers a robust and interpretable solution for detecting AI-generated images by combining physical sensor fingerprinting with deep learning-based pattern analysis. The system leverages Photo-Response Non-Uniformity (PRNU)—a unique sensor-specific noise pattern inherent to every digital camera—to establish the provenance of an image. Since AI-generated images lack such optical signatures, deviations in noise distribution can serve as definitive indicators of synthetic origin.

The architecture comprises three major stages. The first stage performs noise residual extraction using wavelet-based denoising filters to isolate the PRNU signal from image content. The second stage employs a Convolutional Neural Network (CNN) trained on genuine and synthetic PRNU residuals to identify characteristic discrepancies. The third stage integrates Grad-CAM visualization to highlight areas where the model detects inconsistencies, providing interpretable evidence of image authenticity.

To enhance resilience against compression and scaling, the system incorporates an adaptive preprocessing pipeline that normalizes image resolution and illumination. A hybrid loss function combining binary cross-entropy and texture similarity ensures that both classification accuracy and PRNU consistency are optimized during training. This allows the model to generalize effectively across different datasets and generative models, including GANs and diffusion-based architectures.

VI MATERIALS AND METHODS

The Dataset

The dataset used in this work for training and testing is composed of a collection of images divided into two groups (or classes): AI-generated and real camera photographs. First, AI-generated images were created by authors using three different engines: DALL E,

Stable Diffusion, and OpenArt. These images were visually checked to discard those that were not photorealistic. Second, real photos were selected randomly from image databases. There are images from the Dresden Image Database, from the VISION dataset, and also from authors' provided images that were already used in previous studies. There are real photos from the following cameras: Canon Ixus 70 (two instances), Casio Ex Z150 (two instances), Canon PhotoSmart SX720, Canon EOS 1100D, Kodak M1063 (two instances), and Sony ILCE 5000. Photos from smartphones are also included: Huawei P20, Huawei P9, Samsung Galaxy S3 Mini, Apple iPhone 4s, Apple iPhone 5c, Apple iPhone 6, and LG D290.

PRNU Extraction

As the name, Photo Response Non-Uniformity, indicates, PRNU comes from the different light sensitivity of the different pixels (elementary sensors). This is an unavoidable characteristic due to manufacturing imperfections, and it is present on all image sensor chips. PRNU is seen as a multiplicative noise that responds to the following equation:

$$Im_{out} = (I_{ones} + Noise_{cam}).Im_{in} + Noise_{add}$$

ELA Error Level Analysis

ELA pattern is computed to detect irregular distributions of quantization noise. This is a tool normally used to detect image editing. An ELA pattern is normally computed by coding the whole image with JPEG standard at a known, constant, and normally high- quality level (a typical value is 95%); then, the decoded image from the JPEG bit stream is subtracted from the original image.

$$ELA_{img} = img - JPEG^{-1}[JPEG(img, 95\%)]$$

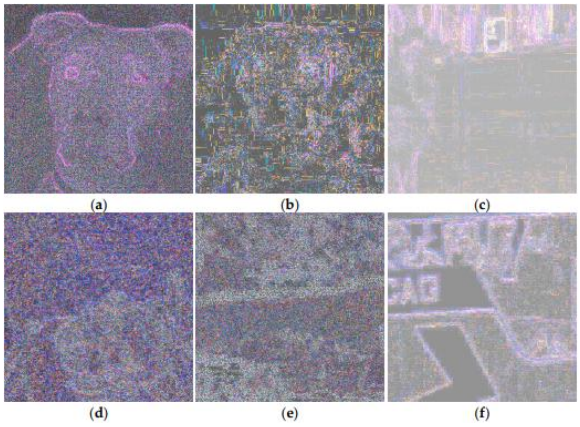

Fig: ELA patterns computed for AI images of Figure 1. (d–f) are examples of ELA patterns for real i

CNNs—Convolutional Neural Networks

CNNs are a cascade of convolutional (or linear filtering) stages accompanied by others of non-linear activation, normalization, and decimation. These stages extract high-level features from low-level data (pixels), so CNNs can process images directly with no need for feature extraction. The initial image is repeatedly filtered and decimated, creating a set of several small images that are finally processed by a classical perceptron (fully connected) stage to obtain the final result. This final result is a numerical vector of as many components as classes to be recognized. The Softmax normalization (the most frequently used at the final stage of CNNs) makes vector coefficients lie in the range Figure 3. (a–c), ELA patterns computed for AI images of Figure 1. (d–f) are examples of ELA patterns for real images. 2.4. CNNs—Convolutional Neural Networks CNNs are a cascade of convolutional (or linear filtering) stages accompanied by others of non-linear activation, normalization, and decimation. These stages extract high-level features from low-level data (pixels), so CNNs can process images directly with no need for feature extraction. The initial image is repeatedly filtered and decimated, creating a set of several small images that are finally processed by a classical perceptron (fully connected) stage to obtain the final result. This final result is a numerical vector of as many components as classes to be recognized. The Softmax normalization (the most frequently used at the final stage of CNNs) makes vector coefficients lie in the range of 0.0–1.0, and, in addition, they always add up to 1.0. The maximum component defines which one is the recognized class.

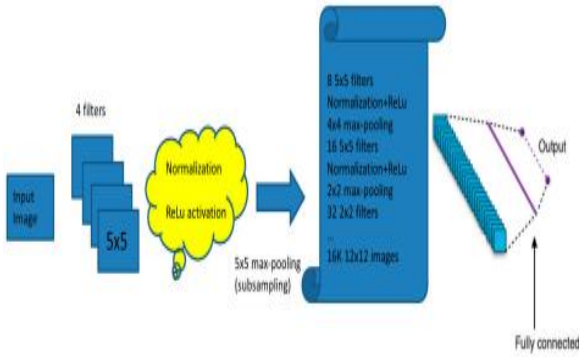

Fig: CNN Structure

VII SYSTEM REQUIREMENTS

SOFTWARE REQUIREMENTS:
Operating system - Windows 7 Ultimate.
Coding Language & Front End - Python.
Back-End - Django-ORM
Designing - Html, CSS, JavaScript.
Data Base - MySQL (WAMP Server).

H/W REQUIREMENTS :-
Processor - Pentium –IV
RAM - 4 GB (min)
Hard Disk - 20 GB
Key Board - Standard Windows Keyboard
Mouse - Two or Three Button Mouse
Monitor – SVGA

VIII RESULTS

CNN nets were trained and tested for both types of feature extraction. This process produces learning curves displayed in Figures 5 and 6. In both cases, a good result is achieved: accuracy is 0.95 for PRNU and 0.98 for ELA. Both trainings have been done with 100 epochs. Training time is bigger for the ELA case (167 minutes versus 109), this is reasonable because ELA images are color ones with three times more information. Blue curves in both figures are the accuracy values obtained for each iteration (measured on the training samples), black curves are accuracy values for the validation set at each epoch (an epoch is equal to n iterations, with n=3 in this case). The curves below (brown and black) are the mean square error (over training and validation set), this is other method for controlling learning. In both cases, the fact that black curves follow the evolution of blue/brown curves demonstrates that neural net is generalizing. In the case of overfitting, the blue curve can go high but the black curve would remain low.

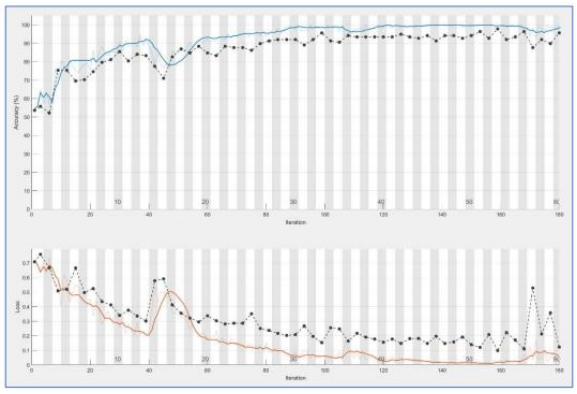

Fig: CNN training for PRNU patterns.

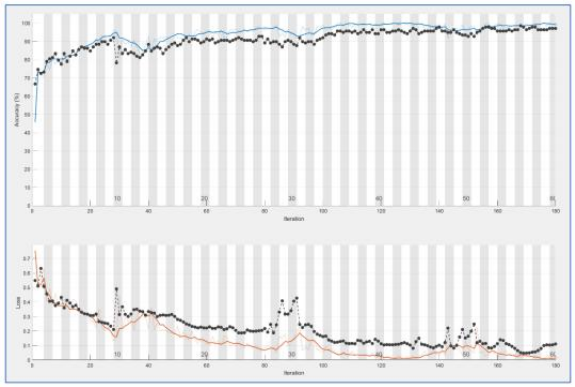

Fig: CNN training for ELA patterns.

IX CONCLUSION

In this paper, we presented a robust approach for detecting AI-synthesized images through the integration of source camera fingerprinting. Our methodology leverages a pretrained ResNet101-based feature extractor initially designed for source camera identification, demonstrating its versatility in discerning between real and AI-generated images. The use of global fingerprints extracted from mid-frequency bands, coupled with the elimination of fragile high-frequency information, reveal that the proposed method consistently achieves high accuracy, particularly excelling in protocols that involve post-processing, such as the anti-forensics attack. This suggests that the mid-frequency features, which our method relies on, remain robust against various forms of image manipulation intended to obfuscate synthetic origins. The generalization capability of our method was rigorously evaluated through cross-model testing, revealing its ability to maintain high accuracy across different datasets and generative models. This underscores the method's potential effectiveness in real-world scenarios where the generative model may not be known beforehand. Achieving near-perfect AUC scores on post-processed images and effectively distinguishing between generative models, our method proves its adaptability and reliability in various domains, including forensics and content moderation, contributing to the ongoing efforts in addressing the challenges posed by synthetic media.

REFERENCES

[1] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, "Analyzing and improving the image quality of stylegan," in Proceedings of the IEEE/CVF

conference on computer vision and pattern recognition, 2020, pp. 8110–8119.

[2] Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila, "Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis," arXiv preprint arXiv:2301.09515, 2023. [3] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," arXiv preprint arXiv:1710.10196, 2017.

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan, "Large scale gan training for high fidelity natural image synthesis," arXiv preprint arXiv:1809.11096, 2018.

[5] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," in International conference on learning representations, 2016.

[6] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al., "Learning transferable visual models from natural language supervision," in International conference on machine learning. PMLR, 2021, pp. 8748–8763.

[7] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever, "Zero-shot text-to-image generation," in International Conference on Machine Learning. PMLR, 2021, pp. 8821–8831.

[8] Siwei Lyu, "Deepfake detection: Current challenges and next steps," in 2020 IEEE international conference on multimedia & expo workshops (ICMEW). IEEE, 2020, pp. 1–6.

[9] Xin Wang, Hui Guo, Shu Hu, Ming-Ching Chang, and Siwei Lyu, "Gangenerated faces detection: A survey and new perspectives," arXiv preprint arXiv:2202.07145, 2022.

[10] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros, "Cnn-generated images are surprisingly easy to spot... for now," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 8695–8704.

[11] Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu, "Multi-attentional deepfake detection," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 2185–2194.

[12] DiegoGragnaniello, Davide Cozzolino, Francesco Marra, Giovanni Poggi, and Luisa Verdoliva, "Are gan generated images easy to detect? a critical analysis of the state-of-the-art," in 2021 IEEE international conference on multimedia and expo (ICME). IEEE, 2021, pp. 1–6.

[13] NingYu,Larry SDavis, and Mario Fritz, "Attributing fake images to gans: Learning and analyzing gan fingerprints," in Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 7556–7566.

[14] LucyChai, David Bau, Ser-Nam Lim, and Phillip Isola, "What makes fake images detectable? understanding properties that generalize," in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16. Springer, 2020, pp. 103–120.

[15] Zhengzhe Liu, Xiaojuan Qi, and Philip HS Torr, "Global texture enhance ment for fake face detection in the wild," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 8060 8069.