Khulna University of Engineering & Technology
Department of Computer Science and Engineering

Topic: Fake Image Detection
CSE 4120: Technical Writing & Seminar

Instructed by:
Dr. K. M. Azharul Hasan
Professor,
Department of CSE, KUET
Sunanda Das
Assistant Professor,
Department of CSE, KUET

Submitted by:
Nushrat Tarmin Meem
Roll: **1907083**
Date of Submission: 03.06.2024

Contents

1 Introduction **5**

2 Background/Problem Statement **6**
2.1 Background 6
2.2 Problem Statement 6

3 Review of Literature **7**

4 Methodology **8**
4.1 Paper 1: New Finding and Unified Framework for Fake Image Detection 8
4.2 Paper 2: Towards Universal Fake Image Detectors that Generalize Across Generative Models 9
4.3 Paper 3: Fake-image detection with Robust Hashing 10
4.4 Comparison of Three Papers 11

5 Result Analysis **12**

6 Findings and Recommendations **15**
6.1 Findings 15
6.2 Recommendations 16

7 Addressing Course Outcomes and Program Outcomes **17**

8 Addressing Complex Engineering Activities **18**

9 Conclusion **20**

10 Publication Details **23**

List of Tables

5.1 Quantitative evaluation results on test 1000 dataset 14

List of Figures

4.1 NAFID network for face forgery detection (Paper 1) 8
4.2 Towards Universal Fake Image Detectors Nearest neighbors method for real-vs-fake image classification (Paper 2) 9
4.3 Robust Hashing Method (Paper 3) 10

5.1 Before and after distributions of real/fake features visualized by t-SNE using NFE module (Paper 1) 12
5.2 Before and after distributions of real/fake features visualized by t-SNE of feature space of different image encoders using nearest neighbors (Paper 2) 13
5.3 Some Outputs from Robust Method 14

10.1 Publication details of three papers 23

Abstract

Now a days, due to excessive use of internet, a major concern has been detected in case of social media along with security issues for proper identification. Besides this, fake images or AI generated images have already taken a large portion on the internet for which it's a very difficult problem in fields like journalism, forensics, and social media. To solve this issue, in total three papers from different sources have been studied regarding this and compared with each other for driving a conclusion to find out which one is the optimal solution for detecting fake images according to accuracy and also some parameter comparison of image processing and computer vision techniques. Different networks including NAFID [3] , CLIP:ViT [7] , Robust Hashing [10] methods have been compared to each other. Above all, future research directions have been discussed focusing on the need for larger and more diverse datasets, real-time detection capabilities and the development of methods that can adapt to the evolving techniques of image manipulation and most importantly can detect AI or GAN generated images.

Chapter 1

Introduction

The production and manipulation of phony photographs has grown in sophistication in recent years, creating serious problems for forensics, social media, and journalism, among other fields. The demand for efficient detection systems grows as realistic false picture creation capabilities advance. For journalism to deliver accurate news reporting, picture authenticity is crucial. Fake photos have the potential to mislead people and erode public confidence in media outlets. In forensics, the capacity to discriminate between authentic and altered photographs is essential to maintaining the integrity of investigations and the administration of justice. Fake photos can spread quickly and extensively on social media platforms, affecting public opinion and disseminating false information. This can be quite harmful.

Multiple sophisticated techniques for detecting phony images have been developed in order to overcome these problems. Neural Architecture for Fake picture Detection (NAFID) [3] is one such technique that uses deep learning to automatically detect anomalies in picture data. Through the examination of an image's underlying features, NAFID [3] is able to identify forgeries that the human eye could miss. On the otherhand, CLIP-ViT [7] , or Contrastive Language-Image Pre-training mixed with Vision Transformers, is another innovative technique. This method offers an effective framework for deciphering and examining the connection between visuals and their written explanations. Through the integration of both textual and visual data, CLIP-ViT [7] improves and broadens the detection process. It can detect differences between the text that goes with an image and the content of the image itself, which is especially helpful for identifying deepfakes and other complex image manipulations. Lastly robust hashing [10] methods provide an alternative by producing distinct hashes, or digital fingerprints, for each image. By comparing photos, these hashes make it possible to find any changes or duplication. This technique works very well for confirming the legitimacy of photos and making sure they haven't been altered since they were taken.

By integrating these advanced techniques—NAFID [3], CLIP-ViT [7], and robust hashing [10] —the accuracy and reliability of fake image detection can be significantly improved. These methods complement each other, providing a multi-faceted approach to detecting fake images.

Chapter 2

Background/Problem Statement

2.1 Background

The development of advanced digital technology has completely changed how we produce, edit, and distribute photos. Digital image manipulation is a serious danger to a number of industries, including social networking, forensics, and journalism. In the field of journalism, the veracity of visual content is crucial to the trustworthiness of news stories. Forensics relies heavily on picture evidence integrity to ensure that investigations are accurate and that justice is served. The quick sharing and consumption of photos on social media platforms makes them especially susceptible to the propagation of false information via phony photographs.

2.2 Problem Statement

Fake image identification is a complicated and serious topic. The volume of photos created and exchanged every day, along with the growing sophistication of image alteration techniques, make traditional methods of image verification, including expert manual examination, insufficient. Deepfakes and other contemporary fake images are produced with sophisticated machine learning algorithms that yield remarkably lifelike results, rendering them undetectable to the unaided eye.

Finding minute irregularities in the image data that point to tampering is one of the main problems in false image detection. Furthermore, reliable techniques are required to continuously confirm the legitimacy of photos, guaranteeing that any modifications are precisely detected. In order to tackle these issues, scientists have created a number of sophisticated techniques for identifying phony images. To sum up, the identification of counterfeit photographs is a serious problem that necessitates sophisticated, automated solutions in order to guarantee the legitimacy of visual material in a variety of contexts. In this continuous fight against image manipulation, the creation and application of methods like NAFID, CLIP-ViT, and strong hashing represent important advancements.

Chapter 3

Review of Literature

Numerous studies have been conducted in the realm of fake image detection, each focusing on different aspects and techniques to identify forgeries. Some works have concentrated on detecting forgery clues in facial features, head poses, blinking patterns, or skin colors. For instance, studies in this field have demonstrated that irregularities in these components may serve as a red flag for photos that have been altered [4] [11] [12]. These investigations use small, sometimes undetectable differences in head movements, atypical skin tone distributions, odd blinking patterns, and facial expression variances [17] [1] to detect bogus photos.

Apart from these physical and facial clues, additional study endeavors have focused on identifying conspicuous alterations in photographs. Observable distortions or alterations that can be found using conventional image processing techniques may be among these changes [12] [2]. These techniques sometimes entail comparing an image to well-known benchmarks or identifying anomalies that may indicate modification by applying basic heuristics.

Still, detecting photographs with slight distortions is one of the major hurdles in bogus image detection. Subtle modifications require more advanced procedures to detect, whereas blatant changes can be relatively easier to discover. Studies have brought attention to the shortcomings of existing detection techniques in these kinds of situations, highlighting the necessity for more sophisticated and sensitive procedures [6] [8] [14] [13].

Modification of texture information has been another important area of attention. Researchers have created techniques to identify texture changes in photos, which can be a clear indicator of image manipulation. These methods seek to identify changes that might otherwise go undetected by examining the regularity and authenticity of texture patterns [16] [13].

The detection of Generative Adversarial Network (GAN) or AI-generated fake images presents an even greater challenge. GANs can produce highly realistic images that are difficult to distinguish from genuine ones. Research in this domain has been dedicated to developing advanced algorithms and techniques specifically designed to detect GAN-generated fakes, recognizing the unique patterns and anomalies that these AI-generated images exhibit [6] [9] [5] [15].

These various approaches collectively contribute to the advancement of fake image detection, addressing the complexities and evolving nature of image manipulation.

Chapter 4

Methodology

4.1 Paper 1: New Finding and Unified Framework for Fake Image Detection

Detecting fake images requires a multifaceted approach, blending traditional image analysis methods with cutting-edge machine learning techniques. First, a heterogeneous dataset comprising real and artificial images is put together. The photos are standardized and any possible interference is eliminated through preprocessing. After that, features are extracted from the images in order to identify statistical, structural, and semantic properties. Based on these features, machine learning models—which include supervised and unsupervised algorithms—are then used to learn and distinguish between authentic and fraudulent images.

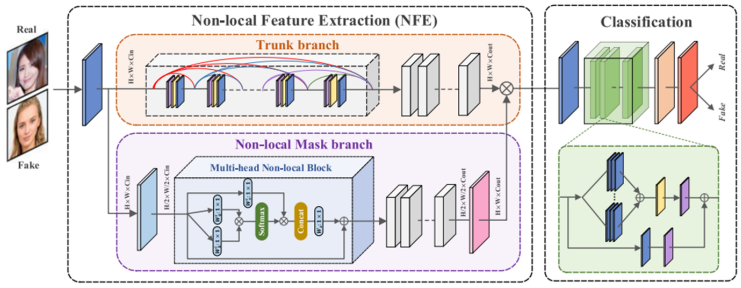


Figure 4.1: NAFID network for face forgery detection (Paper 1)

The overall methodology of this paper has been shown within the diagrams of figure 4.2. It is basically understanding how fake images are created—specifically, through face generation and manipulation techniques using Generative Adversarial Networks (GANs) [6] [15] — motivates the suggested method for detecting fake images, known as NAFID. The approach takes advantage of the finding that, as a result of the generation process, GAN-generated fake faces frequently show stronger non-local self-similarity than real faces by examining their features. As a result of their stronger non-local self-similarity, fake faces typically have higher Peak Signal-to-Noise Ratio (PSNR) values after

denoising. Additionally, the PSNR distribution of fake faces is more concentrated than that of actual faces, indicating less fluctuation. The NFE module is composed of a non-local mask branch that uses multi-head attention to capture various non-local information and a trunk branch for feature extraction. For the purpose of final detection, the collected non-local features are then input into a ResNeXt classification network. This method uses the unique features of GAN-generated [6] [15] fake faces in contrast to real ones to show how easy yet effective it can be to identify phony photos

4.2 Paper 2: Towards Universal Fake Image Detectors that Generalize Across Generative Models

In order to detect fake images, this paper has proposed a methodology that makes use of a feature space that has not been specifically trained for real or fake image classification. The approach does away with the need to train a neural network to discriminate between real and fake classes by utilizing a pre-trained CLIP visual encoder that has been exposed to a sizable dataset of image-text pairs. This encoder was chosen for its ability to capture low-level image details and its wide exposure to a variety of visual content. This paper makes use of the nearest neighbor and linear classification techniques.

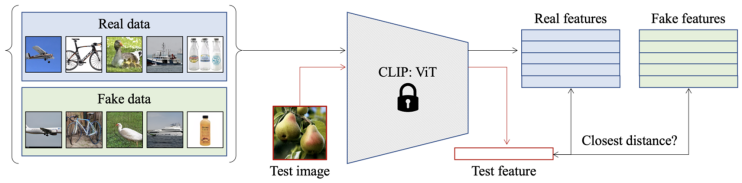


Figure 4.2: Towards Universal Fake Image Detectors Nearest neighbors method for real-vs-fake image classification (Paper 2)

The linear classification method involves layering a single linear layer with sigmoid activation on top of the CLIP encoder. Only this new classification layer is trained using binary cross-entropy loss for binary real-vs-fake classification. While being more computationally efficient, this method preserves the benefits of the nearest neighbor method by training a limited number of parameters in the linear layer. All things considered, the methodology presents a promising approach for fake image detection by using an untrained feature space derived from CLIP to distinguish between real and fake images.

4.3 Paper 3: Fake-image detection with Robust Hashing

This paper has suggested a technique for robust hashing-based fake-image detection that provides an easy-to-use yet efficient way to discriminate between actual and altered photos. Using a particular hashing technique, the methodology computes resilient hash values from reference photos and stores them in a database. In a similar manner, hash values from query photos are calculated and compared to database information. Whether a picture is considered legitimate or false depends on how far its hash value differs from those in the database. This work employs a robust hashing method that consists of several steps: image scaling, rich feature extraction, Gaussian low-pass filtering, and bit string output as the hash value.

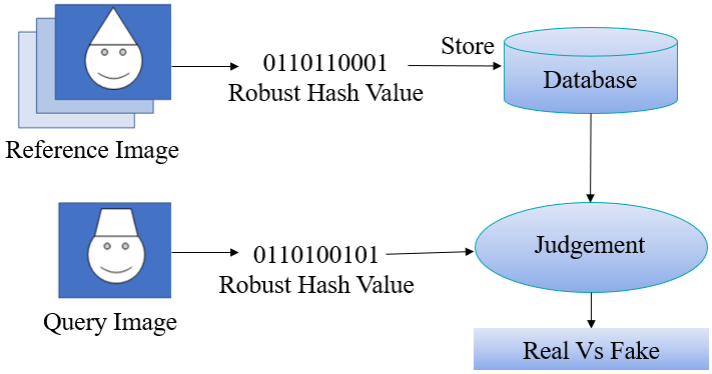


Figure 4.3: Robust Hashing Method (Paper 3)

The experimental evaluation made use of four fake-image datasets: the Image Manipulation Dataset, UADFV, CycleGAN, and StarGAN [6] [15]. JPEG compression was used for all query photos, and a threshold value was selected based on Equal Error Rate performance. The proposed method was compared with Wang et al.'s state-of-the-art fake detection method, which focuses on CNN-generated image identification. The experimental results demonstrate that the suggested method works better than Wang's method in terms of Average Precision (AP) and Accuracy (fake), particularly when working with datasets that entail picture change. All things considered, the proposed method exhibits potential for detecting fake images and offers protection against various forms of picture manipulation.

4.4 Comparison of Three Papers

Paper 1: Non-local Attention based Fake Image Detection (NAFID) network

- Non-local features are extracted using the NFE module
- Probability density function (PDF) of PSNR values after denoising for real and fake images are calculated
- ResNeXt is adopted as the classification network to distinguish real and fake images

Paper 2: CLIP:ViT network (Visual Transformer within Contrastive Language-Image Pre-training [7])

- Feature Space Selection using ViT-L/14 with patch size 14x14
- Cosine distance is calculated to find the nearest neighbors to both real and fake feature banks
- Two simple classification methods are investigated: nearest neighbor and linear probing

Paper 3: Robust Hashing Method (robust hash value stored into database)

- Robust hash value calculated by resizing images to 128x128 pixels, applying 5x5 Gaussian low-pass filtering
- Hamming distance calculation between their hash strings
- Thresholding to determine if the query image is fake or not

Chapter 5

Result Analysis

From paper 1 network; NAFID [3], detects phony faces. Compared to state-of-the-art approaches, NAFID outperforms them on a variety of forgeries datasets with impressive detection accuracy, particularly on datasets featuring complex GAN structures such as StyleGAN2. NAFID's efficacy stems from its capacity to efficiently extract non-local features, as evidenced by the enhancement it confers upon previous models for forgery detection.

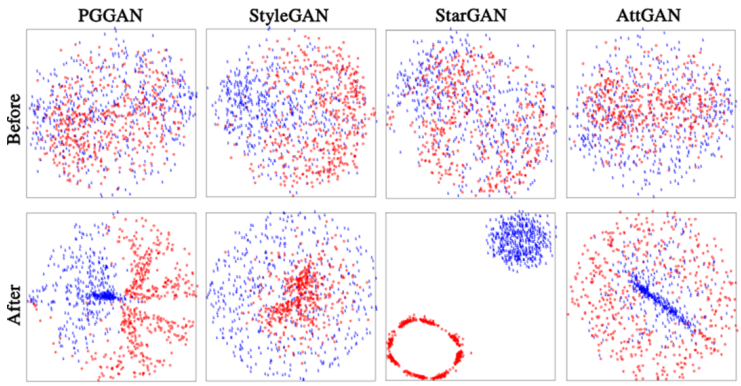


Figure 5.1: Before and after distributions of real/fake features visualized by t-SNE using NFE module (Paper 1)

The usefulness of the non-local feature extraction module in distinguishing between actual and fake images is further confirmed by the visualization results obtained using t-SNE and Grad-CAM. The design and success of NAFID are based on a fundamental insight: fake faces have stronger non-local self-similarity than real ones.

Paper 2 has evaluated several approaches to false picture detection, emphasizing the transferability of these approaches to various generative model types. Conventional deep neural networks have difficulty with generalization and occasionally categorize undetectable false visuals erroneously. But the suggested approach makes use of a pre-trained network's feature space to achieve far higher generalization performance and keeps high accuracy even when used with unidentified generative models. Experiments also reveal that altering the pre-training dataset or backbone design has a considerable impact on the detection efficacy; models trained on a wider variety of datasets perform better.

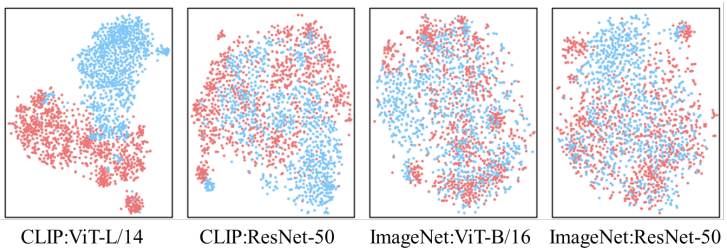


Figure 5.2: Before and after distributions of real/fake features visualized by t-SNE of feature space of different image encoders using nearest neighbors (Paper 2)

Furthermore, the model can still achieve strong generalizability when multiple training data sources, such as ProGAN or LDM, are used. This suggests a latent relationship between different kinds of false pictures. In general, the findings emphasize how crucial it is to use pre-trained feature spaces for reliable false picture identification, particularly in generative model landscapes that are varied and dynamic.

The experimental results for paper 3 demonstrates that the proposed method outperforms Wang's method in terms of Average Precision (AP) and Accuracy (fake) on a range of datasets, including those requiring picture alteration. Specifically, when working with datasets like Image Manipulation Dataset and UADFV, where images are manipulated using methods other than GANs, Wang's method performs significantly worse than the recommended approach. This demonstrates that the recommended approach can withstand a wider range of image modifications.

Figure 5.3: Some Outputs from Robust Method

Furthermore, Wang's method performs poorly, especially when conducting operations like splicing and resizing, while the suggested strategy maintains great accuracy under additional manipulations like JPEG compression, resizing, copy-move, and splicing. This discrepancy stems from Wang's methodology, which focuses on identifying unique characteristics introduced by CNNs and is independent of splicing and resizing processes. Conversely, the robust hashing strategy of the proposed method demonstrates its versatility and effectiveness in detecting fake photographs by enabling it to recognize them using a variety of altering techniques.

Table 5.1: Quantitative evaluation results on test 1000 dataset

Methods	**PGGAN**	**StyleGAN**	**StyleGAN2**
NAFID	100.00	99.68	99.07
CLIP:ViT	-	97.24	-
Robust Hashing	-	-	-
Methods	**StarGAN**	**Deepfakes**	**CycleGAN**
NAFID	100.00	98.74	-
CLIP:ViT	99.60	93.09	99.46
Robust Hashing	100.00	-	100.00

From this table, it is clear that method 1 and method 3 are dominating over method 2 in case of numerical result analysis. Among them method 1 has given better results as it is tested on most of the different sets of test cases.

Chapter 6

Findings and Recommendations

6.1 Findings

Paper 1 Findings: NAFID Network

- Observation on compared to actual faces, artificial faces show more non-local self-similarity
- Suggestion of a non-local feature extraction (NFE) module in light of this observation in order to extract non-local features from fictitious faces
- Method's efficacy is confirmed on multiple forgeries datasets, exhibiting a consistent high performance

Paper 2 Findings: CLIP:ViT Network

- More diverse datasets, such as CLIP, produce better generalization than ImageNet-like datasets
- The training data source has an effect on generalizability; generative models trained on a range of datasets perform better

Paper 3 Findings: Robust Hashing

- When compared to baselines that are currently in use, the suggested robust hashing algorithm performs better at differentiating between actual and fraudulent photos
- When confronted with images from unknown generative models, traditional deep neural networks exhibit notable reductions in accuracy, incorrectly recognizing a large number of bogus images as real

6.2 Recommendations

- In case of NAFID Network, Additional testing on bigger and more varied datasets to evaluate the suggested method's resilience in different situations and investigating different methods or architectures to improve detection performance, particularly in difficult situations such as identifying phony photos from hidden generative models.

- In case of Clip-ViT, how well CLIP scales to handle larger and more diverse datasets, particularly in domains other than picture classification looking for ways to improve CLIP such that its cross-domain generality is preserved when it is used for specific downstream tasks.

- In case of Robust Hashing method, testing the method's robustness by doing tests to see how well it works with various image distortions and modifications and investigating uses for strong hashing techniques that go beyond the identification of false images, such as content-based picture retrieval or digital forensics tamper detection

- In case of NAFID method, evaluation on six fake face datasets with 9K training images which is quite difficult comparing to other methods

- Concerns regarding future image generation technology improvements

- Above all, seeing the result analysis comparing three methods, NAFID method seems to be the optimal solution among them due to accuracy and variability of test cases

Chapter 7

Addressing Course Outcomes and Program Outcomes

Problem Analysis: Problem analysis is used in the seminar lab presentation to thoroughly examine the difficulties in identifying phony photos in a variety of datasets and generative models. Examined are a number of variables that impact the precision and dependability of current detection systems, such as differences in image quality, generative model complexity, and the existence of advanced forgery techniques.
Ethics: Plagiarism-related concerns and ethical considerations are given careful attention in the seminar lab presentation. The topic of debate centers on the moral ramifications of producing and sharing false images, highlighting the possible harm these images may do to people as well as to society. The need of following ethical norms and using appropriate citation techniques is emphasized as a way to guarantee ethical behavior. In this lab, plagiarism is strongly condemned and dealt with. The necessity of upholding originality and academic integrity in research and presentations is brought to the audience's attention. In order to guarantee that any content offered is sourced responsibly and acknowledged, plagiarism prevention techniques are addressed. These include accurately citing references and providing a data summary.
Individual and Teamwork: The value of both individual and team contributions is stressed in the seminar lab presentation. The distinct abilities and knowledge of each member are recognized, and teamwork is encouraged to accomplish shared objectives. In order to promote a harmonious and effective team environment, teamwork techniques including task delegation, effective communication, and conflict resolution are covered.
Communication: In any kind of presentation, clear and succinct information delivery is prioritized as an essential component of effective communication. To make sure that everyone gets the message, emphasis is placed on effective communication techniques like articulation, audience involvement, and active listening. Feedback systems are meant to encourage candid communication and helpful criticism among team members, promoting understanding and ongoing development.

Chapter 8

Addressing Complex Engineering Activities

Handling complicated engineering tasks entails using sophisticated technical expertise and problem-solving techniques to take on challenging tasks across a range of industries. This can include creating complex system designs, streamlining procedures, carrying out in-depth assessments, and coming up with creative fixes in case of complex engineering activities.

Range of Resources:
A variety of tools are used to improve the efficacy of communication, including visual aids like slideshows and charts to support spoken explanations. Written materials are also supplied, such as reports or handouts, to give the audience extensive information and points of reference. Digital platforms can also be used for document or presentation sharing and distant collaboration. These varied materials accommodate various learning preferences and improve audience comprehension and participation during the presentation.

Level of Interaction:
There is a lot of interaction encouraged during the presentation, which motivates the audience to actively participate and become involved. In order to promote communication and idea exchange, there will be opportunities for questions, comments, and discussions throughout the session. Presenters actively solicit input from the audience, requesting that they share their opinions and insights on the topic at hand. Examples of interactive elements that can be used to increase engagement and foster a dynamic learning environment are polls and group activities.

Innovation:
The investigation of original concepts, inventive methods to problem-solving, and the creation of ground-breaking solutions all promote innovation. There is a focus on thinking creatively, questioning accepted wisdom, and expanding the frontiers of technology and knowledge. Diverse viewpoints are encouraged to drive innovation through collaboration and cross-disciplinary communication. Furthermore, a culture of experimentation and risk-taking is fostered to motivate people and groups to investigate novel avenues and grasp prospects for growth.

Concequences for Society and the Environment:
The potential consequences of innovation and technical breakthroughs are carefully addressed in order to guarantee that society and the environment are improved. This entails evaluating how new innovations might affect different stakeholders, such as communities, ecosystems, and the coming generation. Unwanted outcomes that are lessened by legislation include social unrest, environmental destruction, and economic inequality. Sustainable techniques are included into decision-making processes to reduce environmental impact and enhance resource efficiency. In addition, ethical issues influence the adoption and use of technologies to guarantee that they respect social norms and safeguard human rights. Generally speaking, an all-encompassing strategy that strikes a balance between progress and social and environmental obligations is employed to build a more just and sustainable society.

Familarity:
The degree of knowledge or comprehension that people has on a specific subject, idea, or circumstance is referred to as familiarity. It includes the extent to which people are able to identify, comprehend, or feel something. People's levels of familiarity with a subject matter can differ depending on their experiences, education, backgrounds, and exposure to it. It has a significant impact on how people interact with and react to tasks, information, or difficulties, which affects how they make decisions and behave. Gaining more knowledge about a subject often results in an increase in one's self-assurance, skill, and productivity when handling related problems or assignments. And so, familarity with information has been understood throughout this lab.

Chapter 9

Conclusion

In conclusion, a variety of generative models and datasets have demonstrated the effectiveness of the NAFID framework, a revolutionary approach to false image detection. By employing multi-stage classification and non-local feature extraction algorithms, NAFID consistently achieves great levels of accuracy, even on challenging datasets such as StyleGAN2. Its capacity to distinguish between real and fake photos with such accuracy demonstrates how flexible and robust the method is. Similarly, CLIP is a notable invention because of its remarkable generalization capacity, which allows it to cohesively bridge the gap between images and natural words. Comprehending visual and textual inputs is essential for numerous multimodal AI applications, as this ability carries significant benefits. Furthermore, because of its compact and discriminative picture representation, the robust hashing approach is an attractive alternative. It is a useful tool in the toolkit of image authentication techniques due to its resistance to frequent alterations and efficiency in identifying phony photos in a variety of scenarios. Together, these results demonstrate how these techniques could have a big impact on the field of picture forensics and inspire more study to improve and expand their usefulness. Among three different approaches, Non-local Attention based Fake Image Detection (NAFID) network using NFE module is the optimal one. NAFID [3] has (99-100)% accuracy in most of the GAN based test cases and so it works best on GAN or AI generated images Further exploration and validation could enhance the appropriateness of these methods in real-world scenarios.

References

[1] Shruti Agarwal and Hany Farid. Photo forensics from jpeg dimples. In *2017 IEEE Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2017.

[2] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8789–8797, June 2018.

[3] Xin Deng, Bihe Zhao, Zhenyu Guan, and Mai Xu. New finding and unified framework for fake image detection. *IEEE Signal Processing Letters*, 30:90–94, 2023.

[4] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405, 2019.

[5] Yuma Kinoshita and Hitoshi Kiya. Fixed smooth convolutional layer for avoiding checkerboard artifacts in cnns. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3712–3716, 2020.

[6] Francesco Marra, Diego Gragnaniello, Davide Cozzolino, and Luisa Verdoliva. Detection of gan-generated fake images over social networks. *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 384–389, 2018.

[7] U. Ojha, Y. Li, and Y. Lee. Towards universal fake image detectors that generalize across generative models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24480–24489, Los Alamitos, CA, USA, jun 2023. IEEE Computer Society.

[8] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2332–2341, 2019.

[9] Yuan Rao and Jiangqun Ni. A deep learning approach to detection of splicing and copy-move forgeries in images. In *2016 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2016.

[10] Miki Tanaka and Hitoshi Kiya. Fake-image detection with robust hashing. In *2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech)*, pages 40–43, 2021.

[11] Justus Thies, Michael Zollhöfer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2387–2395, 2016.

[12] Chengrui Wang and Weihong Deng. Representative forgery mining for fake face detection. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14918–14927, 2021.

[13] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *ArXiv*, abs/1506.03365, 2015.

[14] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2019.

[15] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6, 2019.

[16] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2242–2251, 2017.

[17] Chenglin Zuo, Ljubomir Jovanov, Bart Goossens, Hiêp Quang Luong, Wilfried Philips, Yu Liu, and Maojun Zhang. Image denoising using quadtree-based nonlocal means with locally adaptive principal component analysis. *IEEE Signal Processing Letters*, 23:434–438, 2016.

Chapter 10

Publication Details

No	Title	Authors	Source	Year
1	New Finding and Unified Framework for Fake Image Detection	Xin Deng, Bihe Zhao, Zhenyu Guan, Mai Xu	IEEE Signal Processing Letters, Vol. 30, 2023	2023
2	Towards Universal Fake Image Detectors that Generalize Across Generative Models	Utkarsh Ojha, Yuheng Li, Yong Jae Lee	IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2023, pp. 24480-24489	2023
3	Fake-image detection with Robust Hashing	Miki Tanaka, Hitoshi Kiya	IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech 2021)	2021

Figure 10.1: Publication details of three papers

All three of the papers have been attached next after this page for more precision.

New Finding and Unified Framework for Fake Image Detection

Xin Deng, *Member, IEEE*, Bihe Zhao, Zhenyu Guan, *Member, IEEE*, and Mai Xu, *Senior Member, IEEE*

***Abstract*—Recently, fake face images generated by generative adversarial network (GAN) have been widely spread in social networks, raising serious social concerns and security risks. To identify the fake images, the top priority is to find what properties make the fake images different from the real images. In this letter, we reveal an important observation about real/fake images, i.e., the GAN generated fake images contain stronger non-local self-similarity than the real images. Motivated by this observation, we propose a simple yet effective non-local attention based fake image detection network, namely NAFID, to distinguish GAN generated fake images from real images. Specifically, we develop a non-local feature extraction (NFE) module to extract the non-local features of the real/fake images, followed by a multi-stage classification module to distinguish the images with the extracted non-local features. Experimental results on various datasets demonstrate the superiority of our NAFID over state-of-the-art (SOTA) face forgery detection methods. More importantly, since the NFE module is independent from classification, we can plug it into any other forgery detection models. The results show that the NFE module can consistently improve the detection accuracy of other models, which verifies the universality of the proposed method.**

***Index Terms*—Fake face detection, generative neural network, non-local similarity.**

I. Introduction

RECENTLY, the face forgery techniques [1], [2], [3], [4], [5], [6], [7], [8] have developed rapidly due to the unprecedented success of generative adversarial network (GAN) [9]. The generated images are so realistic that can even fool the human beings. The high realism of fake faces makes them easily be used for illegal purposes, which has posed serious threats to the social security. To tackle these security issues, it is highly desirable to develop efficient face forgery detection methods. To detect the fake images, the top priority is to find the inherent differences between the fake and real images. Towards this goal, some works [11], [12], [13], [14], [15] focus on detecting the forgery clues in facial features, e.g. head poses, blinking patterns or skin colors. However, these methods can only identify the fake images with obvious changes, but fail to detect those fake images with mild distortion. Recent works [16], [17], [18], [19], [20], [21] have noticed this problem and pay more attention to the low-level subtle details, such as texture information. In addition, there are also some works [22], [23], [24] using prior knowledge from frequency domain to identify the fake face images.

Different from the existing works which rely on carefully designed networks to detect the forgery artifacts, we reveal an important observation about the difference between GAN generated fake faces and real faces. As we know, the non-local self-similarity is an important property of natural images [25], which has been used as prior knowledge to solve many inverse image restoration tasks [26], [27], [28]. Since natural images have non-local property, an intuitive question is whether the GAN generated fake images have the same property. In this letter, through deliberate experiments, we have an important observation, i.e., the fake images contain stronger non-local self-similarity property than the real images. The underlying reason is that the whole or part of the GAN generated images originate from a noise vector with a limited length, which determines that the diversity of the fake images cannot be as rich as the real images. A simple example is shown in Fig. 1, in which we add the same Gaussian noise to the fake/real images and use block-matching and 3D filtering (BM3D) algorithm [10] to denoise the images. Since BM3D achieves denoising by purely exploring the non-local similar patches in the image, the image with stronger non-local similarity tends to have better denoising result. As can be seen, the fake images have higher peak signal-to-noise ratio (PSNR) values than the real images. This observation applies to most GAN based forgery models, including face generation, face swapping, and face editting.

Based on the above observation, we propose a simple yet effective non-local attention based fake image detection network, namely NAFID. The core design of the NAFID network is a non-local feature extraction (NFE) module, where we develop a multi-head non-local block (MNLB) to extract the non-local features from the fake and real images. Then, a multi-stage classification sub-net is used to distinguish the fake and real images based on the extracted non-local features. To evaluate the effectiveness of our framework, we conduct exhaustive experiments on fake face images generated by different generative models. The experimental results show that the proposed method outperforms state-of-the-art face forgery detection methods on various datasets. In addition, since the NFE module is independent from classification, we can incorporate it into other forgery detection models, which has been demonstrated to further improve the detection accuracy of these models.

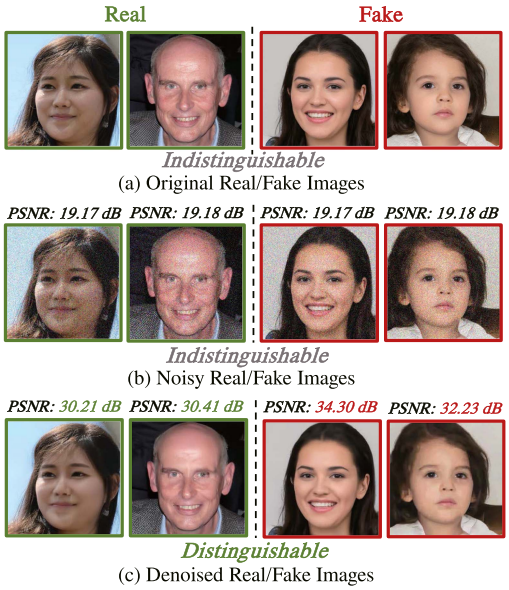
Manuscript received 1 November 2022; revised 11 January 2023; accepted 31 January 2023. Date of publication 9 February 2023; date of current version 15 February 2023. This work was supported in part by NSFC under Grant 62001016 and in part by Young Elite Scientists Sponsorship Program under Grant 2022QNRC001 by CAST. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiangui Kang. *(Xin Deng and Bihe Zhao contributed equally to this work.) (Corresponding author: Zhenyu Guan.)*

Xin Deng, Bihe Zhao, and Zhenyu Guan are with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China (e-mail: cindydeng@buaa.edu.cn; bihezhao@buaa.edu.cn; guanzhenyu@buaa.edu.cn).

Mai Xu is with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China (e-mail: maixu@buaa.edu.cn).

The software code is available through the link https://github.com/BiheZhao/NAFID.

Digital Object Identifier 10.1109/LSP.2023.3243770



(a) Original Real/Fake Images

(b) Noisy Real/Fake Images

(c) Denoised Real/Fake Images

Fig. 1. Illustration about our observation for real/fake images. (a) shows the original real and fake images, which are not distinguishable; (b) shows the real/fake images with Gaussian noise ($\sigma = 30$) added, which are also not distinguishable; (c) shows the denoised real and fake images by BM3D [10], which can be easily distinguishable by the PSNR value.

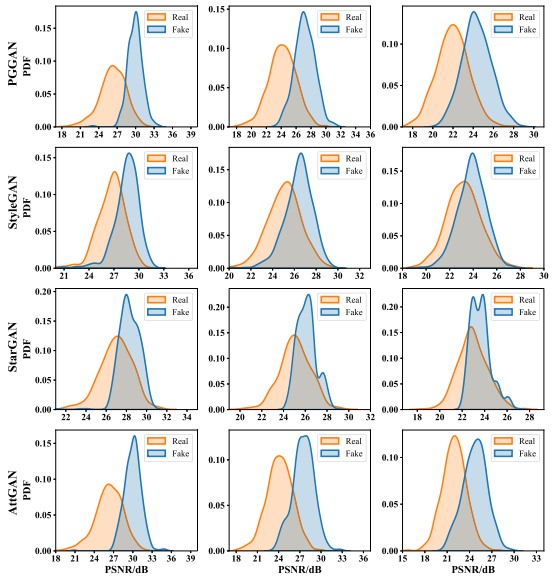


Fig. 2. The probability density function (PDF) of the PSNR for real/fake images denoised by BM3D. The three columns from left to right correspond to the noise level $\sigma = 30, 50$ and 70, respectively.

II. Proposed Method

A. Motivation

To distinguish the GAN generated fake faces from the real faces, we first analyze how these fake images are generated by GAN. There are mainly two types of face forgery methods: face generation and face manipulation. Face generation is a task to synthesize face images from a noise vector [1], [2], [3]. Specifically, the GAN-based face generation methods use a generator to synthesize fake images, and a discriminator to make the generated images indistinguishable. Face manipulation refers to the modification of a real face image, such as face swapping [4], [5] and facial attribute modification [6], [7], [8]. For face manipulation, there is also a controllable noise vector to edit the attributes of the source image. In summary, for both forgery types, the whole or part of the fake images originate from a noise vector with a limited length. In addition, in the process of image generation, there exist several upscaling layers, which enlarge the feature size through either interpolation or copying. All these make the diversity of the generated fake image not as rich as the real image. Therefore, the GAN generated faces could demonstrate more structural patterns and stronger non-local self-similarity than real faces.

Unfortunately, there is no quantitative metric that can directly measure the non-local self-similarity of an image. To tackle this issue, we design an experiment based on BM3D image denoising to evaluate the non-local self-similarity. As we know, the BM3D algorithm achieves image denoising through exploiting the non-local similar patches. The stronger non-local self-similarity of the image leads to the better image denoising result. There are six fake face datasets involved in the experiment, which are generated by the following forgery methods: PGGAN [1], StyleGAN [2], StyleGAN2 [3] belonging to the face generation category, StarGAN v2 [7], Deepfakes [4] and AttGAN [8] belonging to the face manipulation category. We use FaceForensics++ [29] as the Deepfakes dataset. For other methods, the real images are from CelebA [30] and FFHQ [2] datasets. The Gaussian noise with different levels $\sigma = 30, 50$ and 70 are added to the original clean images. After that, we use the BM3D algorithm to denoise both the noisy fake and real images.

Fig. 2 plots the probability density function (PDF) of PSNR values after denoising for both real and fake images. From Fig. 2, we can obtain two intriguing observations: 1) with the same noise level, the denoised fake faces tend to have higher PSNR values than the real faces, and this phenomenon keeps for all the six datasets in different noise levels. This observation reflects that forged faces demonstrate stronger non-local self-similarity than natural faces. 2) the PSNR distribution of fake faces is more concentrated than that of the real images. This indicates that GAN generated faces tend to have less variations and diversities than the real faces. In addition to BM3D, we also carried out the above experiments using the RNAN network [31], and similar observations can be obtained. Due to space limitation, the detailed results are not provided here.

B. Network Architecture

Motivated by the above observations, we propose a a simple yet effective non-local attention based fake image detection network, namely NAFID. We first design a non-local feature extraction (NFE) module to extract non-local features. Then, we use a multi-stage classification sub-net to distinguish real and fake faces based on the extracted non-local features. The architecture of the proposed network is shown in Fig. 3.

1) NFE Module: The NFE module is composed of a trunk branch and a non-local mask branch. The trunk branch consists of several dense blocks [32] for feature extraction. The non-local mask branch includes a multi-head non-local block (MNLB) and several dense blocks to generate non-local attention maps. As shown in Fig. 3, the MNLB replaces single-head attention in non-local block (NLB) [31] with multi-head attention to enhance its representative capability. Each attention head in

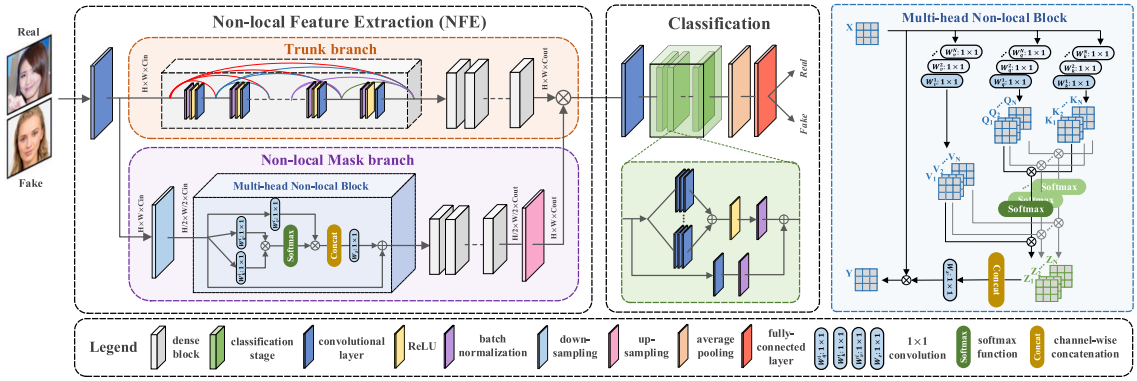


Fig. 3. The proposed NAFID network for face forgery detection. The non-local feature extraction (NFE) module extracts non-local features and the classification module distinguishes the real/fake faces based on extracted non-local features. We use only one trunk branch and one non-local mask branch in NFE module.

MNLB aims at collecting different non-local information from the input feature $\boldsymbol{X}$ with the help of non-local operation [33]. The basic form of non-local operation can be defined as follows:

$$g_q = \frac{1}{\mathcal{C}(q)} \sum_{\forall k} f(q,k) v(k), \tag{1}$$

where q is a target position in feature $\boldsymbol{X}$ and k denotes all possible positions in $\boldsymbol{X}$. $f(q,k)$ measures the similarity between q and k, and $v(k)$ is a representation of k. The $\mathcal{C}(q)$ is a normalizing factor that $\mathcal{C}(q) = \sum_{\forall k} f(q,k)$. Following Wang et al. [26], we choose Gaussian embedding function for $f(q,k)$, which makes $\frac{1}{\mathcal{C}(q)} f(q,k)$ a softmax function. In MNLB block, we have several attention heads, and each attention head aims to collect different non-local information of $\boldsymbol{X}$. Following the above descriptions, the output $\boldsymbol{Z}_i$ of the i-th attention head can be formulated as:

$$\boldsymbol{Z}_i = \mathrm{Softmax}(\boldsymbol{Q}_i \boldsymbol{K}_i^T) \boldsymbol{V}_i, \quad \text{where } i \in \{1, 2, .., N\}. \tag{2}$$

In (2), N denotes the total number of attention heads in MNLB. $\boldsymbol{Q}_i, \boldsymbol{K}_i, \boldsymbol{V}_i$ are different representations of input $\boldsymbol{X}$, i.e., $\boldsymbol{Q}_i = \boldsymbol{W}_q^i \boldsymbol{X}$, $\boldsymbol{K}_i = \boldsymbol{W}_k^i \boldsymbol{X}$, and $\boldsymbol{V}_i = \boldsymbol{W}_v^i \boldsymbol{X}$. The $\boldsymbol{W}_q^i$, $\boldsymbol{W}_k^i$ and $\boldsymbol{W}_v^i$ are weight matrices to be learned. Afterwards, the outputs of all attention heads are concatenated to go through a weight matrix $\boldsymbol{W}_z$ to yield the output $\boldsymbol{Y}$ of MNLB, as follows:

$$\boldsymbol{Y} = \boldsymbol{W}_z \mathrm{Concat}(\boldsymbol{Z}_1, \boldsymbol{Z}_2, \ldots, \boldsymbol{Z}_N) \oplus \boldsymbol{X}, \tag{3}$$

where $\mathrm{Concat}(\cdot)$ denotes channel-wise concatenation and $\oplus$ denotes element-wise addition.

As shown in Fig. 3, since we apply a 2-stride convolutional layer before the MNLB block for computational concerns, we further upsample the output of MNLB to the original size through several dense blocks and bilinear interpolation. This finishes the description of the non-local mask branch. Finally, we apply element-wise multiplication on the outputs of the trunk and non-local mask branch, to extract the non-local features from the input image. Note that the NFE module is flexible, which can be combined with any classification networks, to help improve their detection accuracy.

2) Classification Module: After extracting the non-local features through the NFE module, we simply adopt ResNeXt [34] as the classification network to distinguish the real and fake images. Note that the main contribution of this paper is the observation about the difference between real and fake images. Thus, we do not pay much attention to the design of classification network. We find that a simple ResNeXt is enough to give us advanced detection accuracy, as demonstrated in the experimental results.

III. Experimental Results and Analysis

A. Experimental Setup

1) Real/forgery Datasets: The effectiveness of our method is evaluated on six fake face datasets as mentioned in Section II-A. Each dataset contains 9 K images for training and 1 K images for testing, with equal numbers of real and fake faces. The training/testing images are split randomly in each dataset.

2) Implementation Details: We use the binary cross entropy (BCE) as the loss function to train our network. The Adam optimizer is adopted with a learning rate of 0.001. The number of attention heads is set to $N = 8$. We apply random horizontal flipping and random cropping to enlarge the training samples. The number of training epochs is 400, the batch size is 14, and the patch size is 128×128. We compare the proposed method with six state-of-the-art forgery detection methods including MesoNet [16], ResNeXt [34], Xception [29], F^3-Net [22], RFM [21] and MAT [20]. For fair comparison, the comparison methods are re-trained using the same training samples as ours.

B. Comparison Against Other Methods

Table I shows the detection accuracy of our NAFID and other forgery detection methods. As shown in this table, our NAFID achieves remarkable results on all the six datasets, especially on StyleGAN2. Many methods achieve good detection results in other datasets, but fail in StyleGAN2 which has more sophisticated GAN architecture. For example, MAT achieves 97.36% detection accuracy in StarGAN v2, but with only 73.36% in StyleGAN2. In contrast, our method achieves consistently high detection accuracy on all datasets, with 99.07% accuracy on StyleGAN2. The reason why our method performs well on different datasets is that we have observed the inherent difference between the real and fake images, which is applicable for different forgery datasets.

To further verify the effectiveness of our observation, we plug our NFE module into other comparison networks such as ResNeXt, MesoNet and Xception, to see whether their performance can be improved with NFE. For each detection network, its number of input channels is adjusted to be the same as that

TABLE I
THE DETECTION ACCURACY (%) OF OUR NAFID AND OTHER METHODS ON DIFFERENT FORGERY DATASETS

Methods	PGGAN	StyleGAN	StyleGAN2
MesoNet [16]	99.95	80.27	68.33
ResNeXt [34]	99.88	98.01	79.71
Xception [29]	99.83	99.36	59.72
F^3-Net [22]	**100.00**	99.59	85.47
RFM [21]	99.80	98.71	98.50
MAT [20]	99.71	**99.79**	73.36
NAFID (ours)	**100.00**	99.68	**99.07**
Methods	StarGAN v2	Deepfakes	AttGAN
MesoNet [16]	99.73	80.51	98.69
ResNeXt [34]	97.01	96.63	99.86
Xception [29]	99.34	95.67	99.54
F^3-Net [22]	99.89	97.48	99.95
RFM [21]	99.85	95.89	99.62
MAT [20]	97.36	97.32	99.74
NAFID (ours)	**100.00**	**98.74**	**99.98**

TABLE II
THE DETECTION ACCURACY (%) OF COMPARISON METHODS WITH OUR NFE MODULE PLUGGED IN

Methods	PGGAN	StyleGAN	StyleGAN2
MesoNet [16]	**99.95**	80.27	68.33
+Our NFE	99.22	**97.61**	**90.11**
ResNeXt [34]	99.88	98.01	82.13
+Our NFE	**100.00**	**99.68**	**99.07**
Xception [29]	99.83	99.36	59.72
+Our NFE	**99.90**	**99.69**	**98.64**
Methods	StarGAN v2	Deepfakes	AttGAN
MesoNet [16]	99.73	80.51	98.69
+Our NFE	**100.00**	**92.64**	**99.89**
ResNeXt [34]	97.01	96.63	99.86
+Our NFE	**100.00**	**98.74**	**99.98**
Xception [29]	99.34	95.67	99.54
+Our NFE	**100.00**	**97.11**	**99.81**

of output channels of our NFE module. As shown in Table II, our NFE module consistently improves the detection accuracy of the comparison methods. Take the MesoNet for example, its detection accuracy is increased from 68.33% to 90.11% on StyleGAN2, and increased from 80.51% to 92.64% on Deepfakes dataset. The similar improvement can be seen in ResNeXt and Xception models. These results verify that our NFE module is able to help improve the forgery detection accuracy of different methods.

C. Visualization Results

To visually show the effectiveness of NFE module, we apply t-SNE [35] algorithm to show the distribution of the real/fake images before and after NFE module. As shown in Fig. 4, the distribution of the original real/fake images is fully mixed with no clear clusters. In contrast, after the NFE module, the features of the real/fake images show obvious clustering effect. This phenomenon keeps for different forgery datasets, which demonstrates that the non-local features extracted by our NFE module are effective in dividing the real and fake images. Besides, since our proposed NFE module attempts to extract non-local features of fake faces, the forged faces that contain stronger non-local self-similarity can be separated from the real ones more explicitly, e.g., StarGAN.

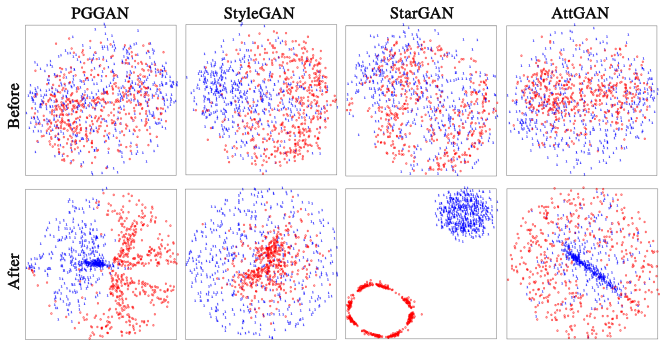


Fig. 4. The distributions of real/fake features visualized by t-SNE before and after the NFE module. The upper figures show the distributions of original real/fake images, and the lower figures show the distributions of the non-local features of real/fake images extracted by NFE module. The real images are denoted by blue dots and fake images are denoted by red ones.

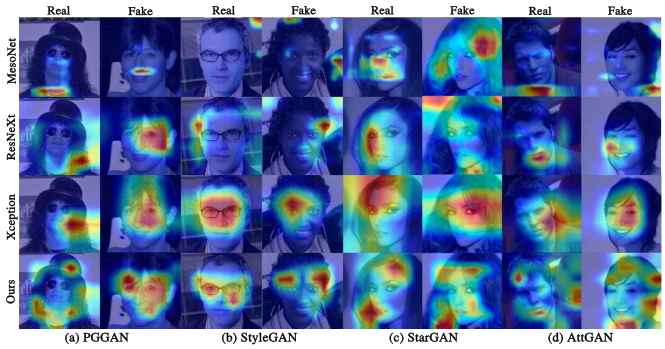


Fig. 5. The Grad-CAM heat maps of our NAFID network and other methods for both real and fake images. The last row shows the results of our method.

To further interpret how the network distinguishes the real from the fake images, we use Grad-CAM algorithm [36] to show the heat maps on the face images. Fig. 5 visualizes the heat maps generated by our NAFID and the comparison methods. As can be seen from the figure, the comparison methods rely on the local regions with forgery artifacts to detect the fake images. Compared to the local attention of the comparison methods, the heat map of our method distributes in different places across the face image. These places possess strong non-local self-similarity, such as hair and skin regions. The visualization results are consistent with the motivation of our network design, i.e., the fake images have stronger non-local properties from the real images.

IV. CONCLUSION

In this letter, we reveal an important observation for fake face detection, i.e., fake faces demonstrate stronger non-local self-similarity than real faces. This observation is demonstrated to be universal to most GAN generated fake images. Based on this observation, we propose a non-local feature extraction module to first extract the non-local features and then use a multi-stage classification module to distinguish the real and fake faces. The effectiveness of the proposed method is verified on various forgery datasets. Moreover, the NFE module is flexible to be plugged into existing forgery detection models to consistently improve their performance.

REFERENCES

[1] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *Proc. Int. Conf. Learn. Representations*, 2018.

[2] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4401–4410.

[3] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8110–8119.

[4] Deepfakes, "Faceswap," Accessed on: 2017. [Online]. Available: https://github.com/deepfakes/faceswap

[5] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of RGB videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 2387–2395.

[6] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8789–8797.

[7] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGAN v2: Diverse image synthesis for multiple domains," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8188–8197.

[8] Z. He, W. Zuo, M. Kan, S. Shan, and X. Chen, "AttGAN: Facial attribute editing by only changing what you want," *IEEE Trans. Image Process.*, vol. 28, no. 11, pp. 5464–5478, Nov. 2019.

[9] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, vol. 27.

[10] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.

[11] Y. Li, M.-C. Chang, and S. Lyu, "In ICTU Oculi: Exposing AI created fake videos by detecting eye blinking," in *Proc. IEEE Int. Workshop Inf. Forensics Secur.*, 2018, pp. 1–7.

[12] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2019, pp. 8261–8265.

[13] U. A. Ciftci, I. Demir, and L. Yin, "FakeCatcher: Detection of synthetic portrait videos using biological signals," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Jul. 15, 2020, doi: 10.1109/TPAMI.2020.3009287.

[14] R. Wang, Z. Yang, W. You, L. Zhou, and B. Chu, "Fake face images detection and identification of celebrities based on semantic segmentation," *IEEE Signal Process. Lett.*, vol. 29, pp. 2018–2022, 2022.

[15] B. Chu, W. You, Z. Yang, L. Zhou, and R. Wang, "Protecting world leader using facial speaking pattern against deepfakes," *IEEE Signal Process. Lett.*, vol. 29, pp. 2078–2082, 2022.

[16] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: A compact facial video forgery detection network," in *Proc. IEEE Int. Workshop Inf. Forensics Secur.*, 2018, pp. 1–7.

[17] Z. Liu, X. Qi, and P. H. Torr, "Global texture enhancement for fake face detection in the wild," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8060–8069.

[18] L. Guarnera, O. Giudice, and S. Battiato, "Deepfake detection by analyzing convolutional traces," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 666–667.

[19] S. Chen, T. Yao, Y. Chen, S. Ding, J. Li, and R. Ji, "Local relation learning for face forgery detection," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, pp. 1081–1088.

[20] H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, and N. Yu, "Multi-attentional deepfake detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 2185–2194.

[21] C. Wang and W. Deng, "Representative forgery mining for fake face detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14923–14932.

[22] Y. Qian, G. Yin, L. Sheng, Z. Chen, and J. Shao, "Thinking in frequency: Face forgery detection by mining frequency-aware clues," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 86–103.

[23] J. Li, H. Xie, J. Li, Z. Wang, and Y. Zhang, "Frequency-aware discriminative feature learning supervised by single-center loss for face forgery detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 6458–6467.

[24] H. Liu et al., "Spatial-phase shallow learning: Rethinking face forgery detection in frequency domain," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 772–781.

[25] M. Zontak and M. Irani, "Internal statistics of a single natural image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 977–984.

[26] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803.

[27] C. Zuo et al., "Image denoising using quadtree-based nonlocal means with locally adaptive principal component analysis," *IEEE Signal Process. Lett.*, vol. 23, no. 4, pp. 434–438, Apr. 2016.

[28] Z. Song, B. Zhong, J. Ji, and K.-K. Ma, "A direction-decoupled non-local attention network for single image super-resolution," *IEEE Signal Process. Lett.*, vol. 29, pp. 2218–2222, 2022.

[29] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++ : Learning to detect manipulated facial images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1–11.

[30] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3730–3738.

[31] Y. Zhang, K. Li, B. Zhong, and Y. Fu, "Residual non-local attention networks for image restoration," in *Proc. Int. Conf. Learn. Representations*, 2019.

[32] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[33] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 2, pp. 60–65.

[34] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1492–1500.

[35] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.

[36] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 618–626.

Towards Universal Fake Image Detectors that Generalize Across Generative Models

Utkarsh Ojha* Yuheng Li* Yong Jae Lee

University of Wisconsin-Madison

Abstract

With generative models proliferating at a rapid rate, there is a growing need for general purpose fake image detectors. In this work, we first show that the existing paradigm, which consists of training a deep network for real-vs-fake classification, fails to detect fake images from newer breeds of generative models when trained to detect GAN fake images. Upon analysis, we find that the resulting classifier is asymmetrically tuned to detect patterns that make an image fake. The real class becomes a 'sink' class holding anything that is not fake, including generated images from models not accessible during training. Building upon this discovery, we propose to perform real-vs-fake classification without learning*; i.e., using a feature space not explicitly trained to distinguish real from fake images. We use nearest neighbor and linear probing as instantiations of this idea. When given access to the feature space of a large pretrained vision-language model, the very simple baseline of nearest neighbor classification has surprisingly good generalization ability in detecting fake images from a wide variety of generative models; e.g., it improves upon the SoTA [50] by* ***+15.07 mAP*** *and* ***+25.90% acc*** *when tested on unseen diffusion and autoregressive models. Our code, models, and data can be found at* https://github.com/Yuheng-Li/UniversalFakeDetect

1. Introduction

The digital world finds itself being flooded with many kinds of fake images these days. Some could be natural images that are doctored using tools like Adobe Photoshop [1, 49], while others could have been generated through a machine learning algorithm. With the rise and maturity of deep generative models [22,29,42], fake images of the latter kind have caught our attention. They have raised excitement because of the quality of images one can generate with ease. They have, however, also raised concerns about their use for malicious purposes [4]. To make matters worse, there is no longer a single source of fake images that needs to be dealt with: for example, synthesized images could take the form of realistic human faces generated using generative adversarial networks [29], or they could take the form of complex scenes generated using diffusion models [42, 45]. One can be almost certain that there will be more modes of fake images coming in the future. With such a diversity, our goal in this work is to develop a general purpose fake detection method which can detect whether any arbitrary image is fake, given access to only one kind of generative model during training; see Fig. 1.

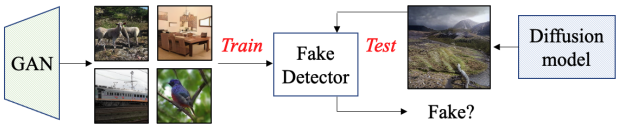


Figure 1. Using images from just one generative model, can we detect images from a different type of generative model as fake?

A common paradigm has been to frame fake image detection as a learning based problem [10, 50], in which a training set of fake and real images are assumed to be available. A deep network is then trained to perform real vs fake binary classification. During test time, the model is used to detect whether a test image is real or fake. Impressively, this strategy results in an excellent generalization ability of the model to detect fake images from different algorithms *within the same generative model family* [50]; e.g., a classifier trained using real/fake images from ProGAN [28] can accurately detect fake images from StyleGAN [29] (both being GAN variants). However, to the best of our knowledge, prior work has not thoroughly explored generalizability across different families of generative models, especially to ones not seen during training; e.g., will the GAN fake classifier be able to detect fake images from diffusion models as well? Our analysis in this work shows that existing methods *do not* attain that level of generalization ability.

Specifically, we find that these models work (or fail to work) in a rather interesting manner. Whenever an image contains the (low-level) fingerprints [25, 50, 52, 53] particu-

*Equal contribution

lar to the generative model used for training (e.g., ProGAN), the image gets classified as fake. *Anything else* gets classified as real. There are two implications: (i) even if diffusion models have a fingerprint of their own, as long as it is not very similar to GAN's fingerprint, their fake images get classified as real; (ii) the classifier doesn't seem to look for features of the real distribution when classifying an image as real; instead, the real class becomes a 'sink class' which hosts anything that is not GAN's version of fake image. In other words, the decision boundary for such a classifier will be closely bound to the particular fake domain.

We argue that the reason that the classifier's decision boundary is unevenly bound to the fake image class is because it is easy for the classifier to latch onto the low-level image artifacts that differentiate fake images from real images. Intuitively, it would be easier to learn to spot the fake pattern, rather than to learn all the ways in which an image could be real. To rectify this undesirable behavior, we propose to perform real-vs-fake image classification using features that are *not trained* to separate fake from real images. As an instantiation of this idea, we perform classification using the *fixed* feature space of a CLIP-ViT [24, 41] model pre-trained on internet-scale image-text pairs. We explore both nearest neighbor classification as well as linear probing on those features.

We empirically show that our approach can achieve significantly better generalization ability in detecting fake images. For example, when training on real/fake images associated with ProGAN [28] and evaluating on unseen diffusion and autoregressive model (LDM+Glide+Guided+DALL-E) images, we obtain improvements over the SoTA [50] by (i) **+15.05mAP and +25.90% acc** with nearest neighbor and (ii) **+19.49mAP and +23.39% acc** with linear probing. We also study the ingredients that make a feature space effective for fake image detection. For example, can we use any image encoder's feature space? Does it matter what domain of fake/real images we have access to? Our key takeaways are that while our approach is robust to the breed of generative model one uses to create the feature bank (e.g., GAN data can be used to detect diffusion models' images and vice versa), one needs the image encoder to be trained on internet-scale data (e.g., ImageNet [21] does not work).

In sum, our main contributions are: (1) We analyze the limitations of existing deep learning based methods in detecting fake images from unseen breeds of generative models. (2) After empirically demonstrating prior methods' ineffectiveness, we present our theory of what could be wrong with the existing paradigm. (3) We use that analysis to present two very simple baselines for real/fake image detection: nearest neighbor and linear classification. Our approach results in state-of-the-art generalization performance, which even the oracle version of the baseline (tuning its confidence threshold on the *test set*) fails to reach. (4) We thoroughly study the key ingredients of our method which are needed for good generalizability.

2. Related work

Types of synthetic images. One category involves altering a portion of a real image, and contains methods which can change a person's attribute in a source image (e.g., smile) using Adobe's photoshop tool [1, 39], or methods which can create DeepFakes replacing the original face in a source image/video with a target face [2, 3]. Another recent technique which can optionally alter a part of a real image is DALL-E 2 [42], which can insert an object (e.g., a chair) in an existing real scene (e.g., office). The other category deals with any algorithm which generates all pixels of an image from scratch. The input for generating such images could be random noise [28, 29], categorical class information [7], text prompts [31, 36, 42, 46], or could even by a collection of images [32]. In this work, we consider primarily this latter category of generated images and see if different detection methods can classify them as fake.

Detecting synthetic images. The need for detecting fake images has existed even before we had powerful image generators. When traditional methods are used to manipulate an image, the alteration in the underlying image statistics can be detected using hand-crafted cues such as compression artifacts [5], resampling [40] or irregular reflections [37]. Several works have also studied GAN synthesized images in their frequency space and have demonstrated the existence of much clearer artifacts [25, 53].

Learning based methods have been used to detect manipulated images as well [15, 44, 49]. Earlier methods studied whether one can even learn a classifier that can detect other images from the same generative model [25, 34, 47], and later work found that such classifiers do not generalize to detecting fakes from other models [19, 53]. Hence, the idea of learning classifiers that generalize to other generative models started gaining attention [17, 35]. In that line of work, [50] proposes a surprisingly simple and effective solution: the authors train a neural network on real/fake images from one kind of GAN, and show that it can detect images from other GAN models as well, if an appropriate training data source and data augmentations are used. [10] extends this idea to detect patches (as opposed to whole images) as real/fake. [6] investigates a related, but different, task of predicting which of two test images is real and which one is modified (fake). Our work analyses the paradigm of training neural networks for fake image detection, showing that their generalizability does not extend to unseen families of generative models. Drawing on this finding, we show the effectiveness of a feature space *not explicitly learned* for the task of fake image detection.

3. Preliminaries

Given a test image, the task is to classify whether it was captured naturally using a camera (real image) or whether it was synthesized by a generative model (fake image). We first discuss the existing paradigm for this task [10, 50], the analysis of which leads to our proposed solution.

3.1. Problem setup

The authors in [50] train a convolutional network (f) for the task of binary real (0) vs fake (1) classification using images associated with one generative model. They train ProGAN [28] on 20 different object categories of LSUN [51], and generate 18k fake images per category. In total, the real-vs-fake training dataset consists of 720k images (360k in *real* class, 360k in *fake* class). They choose ResNet-50 [27] pretrained on ImageNet [21] as the fake classification network, and replace the fully connected layer to train the network for real vs fake classification with the binary cross entropy loss. During training, an intricate data augmentation scheme involving Gaussian blur and JPEG compression is used, which is empirically shown to be critical for generalization. Once trained, the network is used to evaluate the real and fake images from other generative models. For example, BigGAN [7] is evaluated by testing whether its class-conditioned generated images (F_{BigGAN}) and corresponding real images (R_{BigGAN}: coming from ImageNet [21]) get classified correctly; i.e., whether $f(R_{BigGAN}) \approx 0$ and $f(F_{BigGAN}) \approx 1$. Similarly, each generative model (discussed in more detail in Sec. 5.1) has a test set with an equal number of real and fake images associated with it.

3.2. Analysis of why prior work fails to generalize

We start by studying the ability of this network—which is trained to distinguish ProGAN fakes from real images—to detect generated images from unseen methods. In Table 1, we report the accuracy of classifying the real and fake images associated with different families of generative models. As was pointed out in [50], when the target model belongs to the same breed of generative model used for training the real-vs-fake classifier (i.e., GANs), the network shows good overall generalizability in classifying the images; e.g., GauGAN's real/fake images can be detected with 79.25% accuracy. However, when tested on a different family of generative models, e.g., LDM and Guided (variants of diffusion models; see Sec. 5.1), the classification accuracy drastically drops to near *chance* performance![1]

Now, there are two ways in which a classifier can achieve chance performance when the test set has an equal number of real and fake images: it can output (i) a random prediction for each test image, (ii) the same class prediction for all test images. From Table 1, we find that for diffusion models, the classifier works in the latter way, classifying *almost all* images as real regardless of whether they are real (from LAION dataset [48]) or generated. Given this, it seems f has learned an *asymmetric* separation of real and fake classes, where for any image from either LDM (unseen fake) or LAION (unseen real), it has a tendency to disproportionately output one class (real) over the other (fake).

[1]Corresponding precision-recall curves can be found in the appendix.

	CycleGAN	GauGAN	LDM	Guided	DALL-E
Real acc.	98.64	99.4	99.61	99.14	99.61
Fake acc.	62.91	59.1	3.05	4.67	4.9
Average	80.77	79.25	51.33	51.9	52.26
Chance performance	50.00	50.00	50.00	50.00	50.00

Table 1. Accuracy of a real-vs-fake classifier [50] trained on ProGAN images in detecting real and fake images from different types of generative models. LDM, Guided, and DALL-E represent the breeds of image generation algorithms not seen during training.[1]

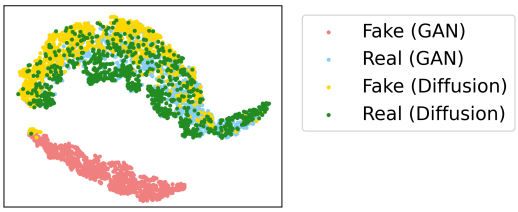


Figure 2. t-SNE visualization of real and fake images associated with two types of generative models. The feature space used is of a classifier trained to distinguish Fake (GAN) from Real (GAN).

To further study this unusual phenomenon, we visualize the feature space used by f for classification. We consider four image distributions: (i) F_{GAN} consisting of fake images generated by ProGAN, (ii) R_{GAN} consisting of the real images used to train ProGAN, (iii) $F_{Diffusion}$ consisting of fake images generated by a latent diffusion model [46], and (iv) $R_{Diffusion}$ consisting of real images (LAION dataset [48]) used to train the latent diffusion model. The real-vs-fake classifier is trained on (i) and (ii). For each, we obtain their corresponding feature representations using the penultimate layer of f, and plot them using t-SNE [33] in Fig. 2. The first thing we notice is that f indeed does not treat real and fake classes equally. In the learned feature space of f, the four image distributions organize themselves into two noticeable clusters. The first cluster is of F_{GAN} (pink) and the other is an amalgamation of the remaining three (R_{GAN} + $F_{Diffusion}$ + $R_{Diffusion}$). In other words, f can easily distinguish F_{GAN} from the other three, but the learned real class does not seem to have any property (a space) of its own, but is rather used by f to form a *sink class*, which hosts anything that is not F_{GAN}. The second thing we notice is that the cluster surrounding the learned fake class is very condensed compared to the one surrounding the learned real class, which is much more open. This indicates that f can detect a common property among im-

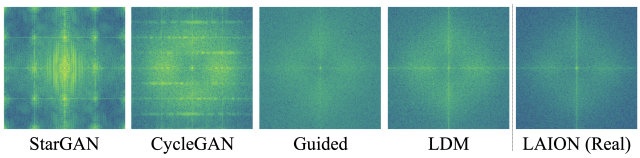


Figure 3. Average frequency spectra of each domain. The first four correspond to fake images from GANs and diffusion models. The last one represents real images from LAION [48] dataset.

ages from F_{GAN} with more ease than detecting a common property among images from R_{GAN}.

But why is it that the property that f finds to be common among F_{GAN} is useful for detecting fake images from other GAN models (e.g., CycleGAN), but not for detecting $F_{Diffusion}$? In what way are fake images from diffusion models different than images from GANs? We investigate this by visualizing the frequency spectra of different image distributions, inspired by [8, 9, 50, 53]. For each distribution (e.g., F_{BigGAN}), we start by performing a high pass filtering for each image by subtracting from it its median blurred image. We then take the average of the resulting high frequency component across 2000 images, and compute the Fourier transform. Fig. 3 shows this average frequency spectra for four fake domains and one real domain. Similar to [50], we see a distinct and repeated pattern in StarGAN and CycleGAN. However, this pattern is missing in the fake images from diffusion models (Guided [23] and LDM [46]), similar to images from a real distribution (LAION [48]). So, while fake images from diffusion models seem to have some common property of their own, Fig. 3 indicates that that property is not of a similar nature as the ones shared by GANs.

Our hypothesis is that when f is learning to distinguish between F_{GAN} and R_{GAN}, it latches onto the artifacts depicted in Fig. 3, learning only to look for the presence/absence of those patterns in an image. Since this is sufficient for it to reduce the training error, it largely ignores learning any features (e.g., smooth edges) pertaining to the *real* class. This, in turn, results in a skewed decision boundary where a fake image from a diffusion model, lacking the GAN's fingerprints, ends up being classified as real.

4. Approach

If learning a neural network f is not an ideal way to separate real ($\mathcal{R}$) and fake ($\mathcal{F}$) classes, what should we do? The key, we believe, is that the classification process should happen in a feature space which has *not been learned* to separate images from the two classes. This might ensure that the features are not biased to recognize patterns from one class disproportionately better than the other.

Choice of feature space. As an initial idea, since we might not want to learn any features, can we simply perform the classification in pixel space? This would not work, as pixel space would not capture any meaningful information (e.g., edges) beyond point-to-point pixel correspondences. So, any classification decision of an image should be made after it has been mapped into some feature space. This feature space, produced by a network and denoted as ϕ, should have some desirable qualities.

First, ϕ should have been exposed to a large number of images. Since we hope to design a general purpose fake image detector, its functioning should be consistent for a wide variety of real/fake images (e.g., a human face, an outdoor scene). This calls for the feature space of ϕ to be heavily populated with different kinds of images, so that for any new test image, it knows how to embed it properly. Second, it would be beneficial if ϕ, while being general overall, can also capture low-level details of an image. This is because differences between real and fake images arise particularly at low-level details [10, 53].

To satisfy these requirements, we consider leveraging a large network trained on huge amounts of data, as a possible candidate to produce ϕ. In particular, we choose a variant of the vision transformer, ViT-L/14 [24], trained for the task of image-language alignment, CLIP [41]. CLIP:ViT is trained on an extraordinarily large dataset of 400M image-text pairs, so it satisfies the first requirement of sufficient exposure to the visual world. Additionally, since ViT-L/14 has a smaller starting patch size of 14 $\times$ 14 (compared to other ViT variants), we believe it can also aid in modeling the low-level image details needed for real-vs-fake classification. Hence, for all of our main experiments, we use the last layer of CLIP:ViT-L/14's visual encoder as ϕ.

The overall approach can be formalized in the following way. We assume access to images associated with a single generative model (e.g., ProGAN, which is the same constraint as in [50]). $\mathcal{R} = \{r_1, r_2, ..., r_N\}$, and $\mathcal{F} = \{f_1, f_2, ..., f_N\}$ denote the real and fake classes respectively, each containing N images. $\mathcal{D} = \{\mathcal{R} \cup \mathcal{F}\}$ denotes the overall training set. We investigate two simple classification methods: nearest neighbor and linear probing. Importantly, both methods utilize a feature space that is entirely untrained for real/fake classification.

Nearest neighbor. Given the pre-trained CLIP:ViT visual encoder, we use its final layer ϕ to map the entire training data to their feature representations (of 768 dimensions). The resulting feature bank is $\phi_{bank} = \{\phi_{\mathcal{R}} \cup \phi_{\mathcal{F}}\}$ where $\phi_{\mathcal{R}} = \{\phi_{r_1}, \phi_{r_2}, ..., \phi_{r_N}\}$ and $\phi_{\mathcal{F}} = \{\phi_{f_1}, \phi_{f_2}, ..., \phi_{f_N}\}$. During test time, an image x is first mapped to its feature representation ϕ_x. Using cosine distance as the metric d, we find its nearest neighbor to both the real ($\phi_{\mathcal{R}}$) and fake ($\phi_{\mathcal{F}}$) feature banks. The prediction—real:0, fake:1—is given based

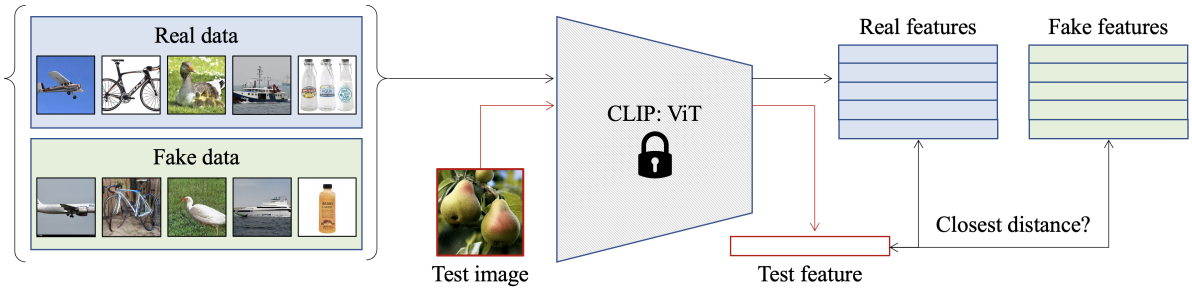


Figure 4. **Nearest neighbors for real-vs-fake classification.** We first map the real and fake images to their corresponding feature representations using a pre-trained CLIP:ViT network *not trained for this task*. A test image is mapped into the same feature space, and cosine distance is used to find the closest member in the feature bank. The label of that member is the predicted class.

on the smaller distance of the two:

$$\text{pred}(x) = \begin{cases} 1, & \text{if } \min_i \left(d(\phi_x, \phi_{f_i})\right) < \min_i \left(d(\phi_x, \phi_{r_i})\right) \\ 0, & \text{otherwise.} \end{cases}$$

The CLIP:ViT encoder is always kept frozen; see Fig. 4.

Linear classification. We take the pre-trained CLIP:ViT encoder, and add a single linear layer with sigmoid activation on top of it, and train *only* this new classification layer ψ for binary real-vs-fake classification using binary cross entropy loss:

$$\mathcal{L} = -\sum_{f_i \in \mathcal{F}} \log(\psi(\phi_{f_i})) - \sum_{r_i \in \mathcal{R}} \log(1 - \psi(\phi_{r_i})).$$

Since such a classifier involves training only a few hundred parameters in the linear layer (e.g., 768), conceptually, it will be quite similar to nearest neighbor and retain many of its useful properties. Additionally, it has the benefit of being more computation and memory friendly.

5. Experiments

We now discuss the experimental setup for evaluating the proposed method for the task of fake image detection.

5.1. Generative models studied

Since new methods of creating fake images are always coming up, the standard practice is to limit access to only one generative model during training, and test the resulting model on images from unseen generative models. We follow the same protocol as described in [50] and use ProGAN's real/fake images as the training dataset.

During evaluation, we consider a variety of generative models. First, we evaluate on the models used in [50]: ProGAN [28], StyleGAN [29], BigGAN [7], CycleGAN [54], StarGAN [13], GauGAN [38], CRN [12], IMLE [30], SAN [18], SITD [11], and DeepFakes [47]. Each generative model has a collection of real and fake images. Additionally, we evaluate on guided diffusion model [23], which is trained for the task for class conditional image synthesis on the ImageNet dataset [21]. We also perform evaluation on recent text-to-image generation models: (i) Latent diffusion model (LDM) [46] and (ii) Glide [36] are variants of diffusion models, and (iii) DALL-E [43] is an autoregressive model (we consider its open sourced implementation DALL-E-mini [20]). For these three methods, we set the LAION dataset [48] as the real class, and use the corresponding text descriptions to generate the fake images.

LDMs can be used to generate images in different ways. The standard practice is to use a text-prompt as input, and perform 200 steps of noise refinement (LDM 200). One can also generate an image with the help of guidance (LDM 200 w/CFG), or use fewer steps for faster sampling (LDM 100). Similarly, we also experiment with different variants of a pre-trained Glide model, which consists of two separate stages of noise refinement. The standard practice is to use 100 steps to get a low resolution image at 64×64, then use 27 steps to upsample the image to 256×256 in the next stage (Glide 100-27). We consider two other variants as well : Glide 50-27 and Glide 100-10 based on the number of refinement steps in the two stages. All generative models synthesize 256×256 resolution images.

5.2. Real-vs-Fake classification baselines

We compare with the following state-of-the-art baselines: (i) Training a classification network to give a real/fake decision for an image using binary cross-entropy loss [50]. The authors take a ResNet-50 [27] pre-trained on ImageNet, and finetune it on ProGAN's real/fake images (henceforth referred as trained deep network). (ii) We include another variant where we change the backbone to CLIP:ViT [24] (to match our approach) and train the network for the same task. (iii) Training a similar classification network on a patch level instead [10], where the authors propose to truncate either a ResNet [27] or Xception [14] (at Layer1 and Block2 respectively) so that a smaller receptive field is considered when making the decision. This method was primarily proposed for detecting generated *facial* images, but

we study whether the idea can be extended to detect more complex fake images. (iv) Training a classification network where input images are first converted into their corresponding co-occurrence matrices [35] (a technique shown to be effective in image steganalysis and forensics [16, 26]), conditioned on which the network predicts the real/fake class. (v) Training a classification network on the frequency spectrum of real/fake images [53], a space which the authors show as better in capturing and displaying the artifacts present in the GAN generated images. All training details can be found in the supplementary.

5.3. Evaluation metrics

We follow existing works [10, 25, 35, 50, 53] and report both average precision (AP) and classification accuracy. To compute accuracy for the baselines, we tune the classification threshold on the held-out training validation set of the available generative model. For example, when training a classifier on data associated with ProGAN, the threshold is chosen so that the accuracy on a held out set of ProGAN's real and fake images can be maximized. In addition, we also compute an upper-bound *oracle* accuracy for [50], where the classifier's threshold is calibrated directly on each test set separately. This is to gauge the best that the classifier can perform on each test set (details in supplementary).

6. Results

We start by comparing our approach to existing baselines in their ability to classify different types of real/fake images, and then study the different components of our approach.

6.1. Detecting fake images from unseen methods

Table 2 and Table 3 show the average precision (AP) and classification accuracy, respectively, of all methods (rows) in detecting fake images from different generative models (columns). For classification accuracy, the numbers shown are averaged over the real and fake classes for each generative model.[2] All methods have access to only ProGAN's data (except [53], which uses CycleGAN's data), either for training the classifier or for creating the NN feature bank.

As discussed in Sec. 3.2, the trained classifier baseline [50] distinguishes real from fakes with good accuracy for other GAN variants. However, the accuracy drops drastically (sometimes to nearly chance performance ∼50-55%; e.g., LDM variants) for images from most unseen generative models, where all types of fake images are classified mostly as real (please see Table C in the supplementary). Importantly, this behavior does not change even if we change the backbone to CLIP:ViT (the one used by our methods). This tells us that the issue highlighted in Fig. 2 affects deep neural networks in general, and not just ResNets. Performing classification on a patch-level [10], using co-occurence matrices [35], or using the frequency space [53] does not solve the issue either, where the classifier fails to have a consistent detection ability, sometimes even for methods within the same generative model family (e.g., GauGAN/BigGAN). Furthermore, even detecting real/fake patches in images from the same training domain (ProGAN) can be difficult in certain settings (Xception). This indicates that while learning to find patterns within small image regions might be sufficient when patches do not vary too much (e.g., facial images), it might not be sufficient when the domain of real and fake images becomes more complex (e.g., natural scenes).

Our approach, on the other hand, show a *drastically better generalization performance* in detecting real/fake images. We observe this first by considering models within the training domain, i.e., GANs, where our NN variants and linear probing achieve an average accuracy of ∼93% and ∼95% respectively, while the best performing baseline, trained deep networks - Blur+JPEG(0.5) achieves ∼85% (improvements of **+8-10%**). This discrepancy in performance becomes more pronounced when considering unseen methods such as diffusion (LDM+Guided+Glide) and autoregressive models (DALL-E), where our NN variants and linear probing achieve 82-84% average accuracy and ∼82% respectively compared to 53-58% by trained deep networks variants [50] (improvements of **+25-30%**). In terms of average precision, the best version of the trained deep network's AP is very high when tested on models from the same GAN family, 94.19 mAP, but drops when tested on unseen diffusion/autoregressive models, 75.51 mAP. Our NN variants and linear probing maintain a high AP both within the same (GAN) family domain, 96.36 and 99.31 mAP, and on unseen diffusion/autoregressive models, 90.58 and 95.00 mAP, resulting in an improvement of about **+15-20 mAP**. These improvements remain similar for our NN variants for voting pool size from k=1 to k=9, which shows that our method is not too sensitive to this hyperparameter.

In sum, these results clearly demonstrate the advantage of using the feature space of a frozen, pre-trained network that is *blind* to the downstream real/fake classification task.

6.2. Allowing the trained classifier to cheat

As described in Sec. 5.3, we experiment with an oracle version of the trained classifier baseline [50], where the threshold of the classifier is tuned directly on each *test set*. Even this flexibility, where the network essentially *cheats(!)*, does not make that classifier perform nearly as well as our approach, especially for models from unseen domains; for example, our nearest neighbor ($k = 9$) achieves an average classification accuracy of 84.25%, which is **7.99% higher than that of the oracle baseline** (76.26%). This shows that the issue with training neural networks for this task is not just the improper threshold at test time. In-

[2] See appendix which further breaks down the accuracies for real/fake.

Detection method	Variant	Generative Adversarial Networks						Deep fakes	Low level vision		Perceptual loss		Guided	LDM			Glide			DALL-E	Total
		Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN		SITD	SAN	CRN	IMLE		200 steps	200 w/ CFG	100 steps	100 27	50 27	100 10		mAP
Trained deep network [50]	Blur+JPEG (0.1)	**100.0**	93.47	84.5	**99.54**	89.49	98.15	89.02	73.75	59.47	98.24	98.4	73.72	70.62	71.0	70.54	80.65	84.91	82.07	70.59	83.58
	Blur+JPEG (0.5)	**100.0**	96.83	88.24	98.29	98.09	95.44	66.27	86.0	61.2	**98.94**	**99.52**	68.57	66.0	66.68	65.39	73.29	78.02	76.23	65.93	81.52
	ViT:CLIP (B+J 0.5)	99.98	93.32	83.63	88.14	92.81	84.62	67.23	**93.48**	55.21	88.75	96.22	55.74	52.52	54.51	52.2	56.64	61.13	56.64	62.74	73.44
Patch classifier [10]	ResNet50-Layer1	98.86	72.04	68.79	92.96	55.9	92.06	60.18	65.82	52.87	68.74	67.59	70.05	87.84	84.94	88.1	74.54	76.28	75.84	77.07	75.28
	Xception-Block2	80.88	72.84	71.66	85.75	65.99	69.25	76.55	76.19	76.34	74.52	68.52	75.03	87.1	86.72	86.4	85.37	83.73	78.38	75.67	77.73
Co-occurence [35]	-	99.74	80.95	50.61	98.63	53.11	67.99	59.14	68.98	60.42	73.06	87.21	70.20	91.21	89.02	92.39	89.32	88.35	82.79	80.96	78.11
Freq-spec [53]	CycleGAN	55.39	**100.0**	75.08	55.11	66.08	**100.0**	45.18	47.46	57.12	53.61	50.98	57.72	77.72	77.25	76.47	68.58	64.58	61.92	67.77	66.21
Ours	NN, $k = 1$	**100.0**	98.14	94.49	86.68	99.26	99.53	**93.09**	78.46	67.54	83.13	91.07	79.31	95.84	79.84	95.97	93.98	95.17	**96.05**	88.51	90.32
	NN, $k = 3$	**100.0**	98.13	94.46	86.67	99.25	99.53	93.03	78.54	67.54	83.13	91.06	79.26	95.81	79.78	95.94	93.94	95.13	94.60	88.47	90.22
	NN, $k = 5$	**100.0**	98.13	94.46	86.66	99.25	99.53	93.02	78.54	67.54	83.12	91.06	79.25	95.81	79.78	95.94	93.94	95.13	94.60	88.46	90.22
	NN, $k = 9$	**100.0**	98.13	94.46	86.66	99.25	99.53	91.67	78.54	67.54	83.12	91.06	79.24	95.81	79.77	95.93	93.93	95.12	94.59	88.45	90.14
	LC	**100.0**	99.46	**99.59**	97.24	**99.98**	99.60	82.45	61.32	**79.02**	96.72	99.00	**87.77**	**99.14**	**92.15**	**99.17**	**94.74**	**95.34**	94.57	**97.15**	**93.38**

Table 2. **Generalization results.** Average precision (AP) of different methods for detecting real/fake images. Models outside the GANs column can be considered as the generalizing domain. The improvements using the fixed feature backbone (Ours NN/LC) over the best performing baseline [50] is particularly noticeable when evaluating on unseen generative models, where our best performing method has significant gains over the best performing baseline: +9.8 mAP overall and +19.49 mAP across unseen diffusion & autoregressive models.

Detection method	Variant	Generative Adversarial Networks						Deep fakes	Low level vision		Perceptual loss		Guided	LDM			Glide			DALL-E	Total
		Pro-GAN	Cycle-GAN	Big-GAN	Style-GAN	Gau-GAN	Star-GAN		SITD	SAN	CRN	IMLE		200 steps	200 w/ CFG	100 steps	100 27	50 27	100 10		Avg. acc
Trained deep network [50]	Blur+JPEG (0.1)	99.99	85.20	70.20	85.7	78.95	91.7	53.47	66.67	48.69	86.31	86.26	60.07	54.03	54.96	54.14	60.78	63.8	65.66	55.58	69.58
	Blur+JPEG (0.5)	**100.0**	80.77	58.98	69.24	79.25	80.94	51.06	56.94	47.73	87.58	94.07	51.90	51.33	51.93	51.28	54.43	55.97	54.36	52.26	64.73
	Oracle* (B+J 0.5)	**100.0**	90.88	82.40	**93.11**	93.52	87.27	62.48	**76.67**	57.04	**95.28**	**96.93**	65.20	63.15	62.39	61.50	65.36	69.52	66.18	60.10	76.26
	ViT:CLIP (B+J 0.5)	98.94	78.80	60.62	60.56	66.82	62.31	52.28	65.28	47.97	64.09	79.54	50.66	50.74	51.04	50.76	52.15	53.07	52.06	53.18	60.57
Patch classifier [10]	ResNet50-Layer1	94.38	67.38	64.62	82.26	57.19	80.29	55.32	64.59	51.24	54.29	55.11	65.14	79.09	**76.17**	79.36	67.06	68.55	68.04	69.44	68.39
	Xception-Block2	75.03	68.97	68.47	79.16	64.23	63.94	75.54	75.14	**75.28**	72.33	55.3	67.41	76.5	76.1	75.77	74.81	73.28	68.52	67.91	71.24
Co-occurence [35]	-	97.70	63.15	53.75	92.50	51.1	54.7	57.1	63.06	55.85	65.65	65.80	60.50	70.7	70.55	71.00	70.25	69.60	69.90	67.55	66.86
Freq-spec [53]	CycleGAN	49.90	**99.90**	50.50	49.90	50.30	**99.70**	50.10	50.00	48.00	50.60	50.10	50.90	50.40	50.40	50.30	51.70	51.40	50.40	50.00	55.45
Ours	NN, $k = 1$	99.58	94.70	86.95	80.24	96.67	98.84	80.9	71.0	56.0	66.3	76.5	68.76	89.56	68.99	89.51	86.44	88.02	87.27	77.52	82.30
	NN, $k = 3$	99.58	95.04	87.63	80.55	96.94	98.77	83.05	71.5	59.5	66.69	76.87	70.02	90.37	70.17	90.57	87.84	89.34	88.78	79.29	83.28
	NN, $k = 5$	99.60	94.32	88.23	80.60	97.00	98.90	83.85	71.5	60.0	67.04	78.02	70.55	90.89	70.97	91.01	88.42	90.07	89.60	80.19	83.72
	NN, $k = 9$	99.54	93.49	88.63	80.75	97.11	98.97	**84.5**	71.5	61.0	69.27	79.21	**71.06**	91.29	72.02	91.29	**89.05**	**90.67**	**90.08**	81.47	**84.25**
	LC	**100.0**	98.50	**94.50**	82.00	**99.50**	97.00	66.60	63.00	57.50	59.5	72.00	70.03	**94.19**	73.76	**94.36**	79.07	79.85	78.14	**86.78**	81.38

Table 3. **Generalization results.** Analogous result of Table 2, where we use classification accuracy (averaged over real and fake images) to compare the methods. Oracle with * indicates that the method uses the test set to calibrate the confidence threshold. The fixed feature backbone (Ours NN/LC) has a significant gain in accuracy (+25-30% over the baselines) when testing on unseen generative model families.

stead, the trained network fundamentally cannot do much other than to look for a certain set of fake patterns; and in their absence, has issues looking for features pertaining to the real distribution. And that is where the feature space of a *model not trained on this task* has its advantages; even when those fake features are absent, there will still be other features useful for classification, which were not learned to be ruled out during the real-vs-fake training process.

6.3. Effect of network backbone

So far, we have seen the surprisingly good generalizability of nearest neighbor / linear probing using CLIP:ViT-L/14's feature space. In this section, we study what happens if the backbone architecture or pre-training dataset is changed. We experiment with our linear classification variant, and consider the following settings: (i) CLIP:ViT-L/14, (ii) CLIP:ResNet-50, (iii) ImageNet:ResNet-50, and (iv) ImageNet:ViT-B/16. For each, we again use ProGAN's real/fake image data as the training data.

Fig. 5 shows the accuracy of these variants on the same models. The key takeaway is that both the network architecture as well as the dataset on which it was trained on play a crucial role in determining the effectiveness for fake image detection. Visual encoders pre-trained as part of the CLIP system fare better compared to those pre-trained on ImageNet. This could be because CLIP's visual encoder gets to see much more diversity of images, thereby exposing it to a much bigger *real distribution* than a model trained on ImageNet. Within CLIP, ViT-L/14 performs better than ResNet-50, which could partly be attributed to its bigger architecture and global receptive field of the attention layers.

We also provide a visual analysis of the pre-trained distributions. Using each of the four model's feature banks consisting of the same real and fake images from ProGAN, we plot four t-SNE figures and color code the resulting 2-D points using binary (real/fake) labels in Fig. 7. CLIP:ViT-L/14's space best separates the real (red) and fake (blue) features, followed by CLIP:ResNet-50. ImageNet:ResNet-50 and ImageNet:ViT-B/16 do not seem to have any proper

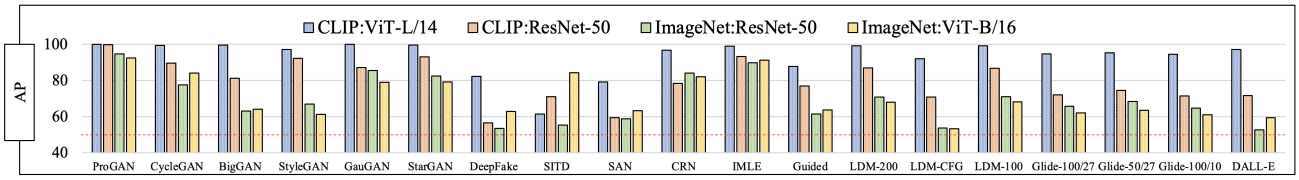


Figure 5. **Ablation on the network architecture and pre-training dataset.** A network trained on the task of CLIP is better equipped at separating fake images from real, compared to networks trained on ImageNet classification. The red dotted line depicts chance performance.

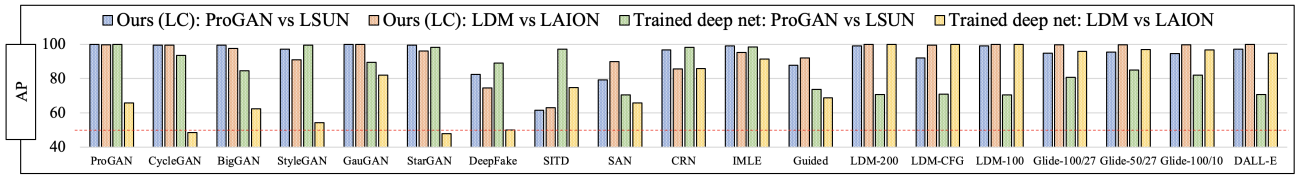


Figure 6. **Average precision of methods with respect to training data.** Both our linear classifier on CLIP:ViT's features and the baseline trained deep network [50] are given access to two different types of training data: (i) $\mathcal{R}$ = LSUN [51] and $\mathcal{F}$ = ProGAN [28], (ii) $\mathcal{R}$ = LAION [48] and $\mathcal{F}$ = LDM [46]. Irrespective of the training data source, our linear classifier preserves its ability to generalize well on images from other unseen generative model families, which is not the case for the baseline trained deep network.

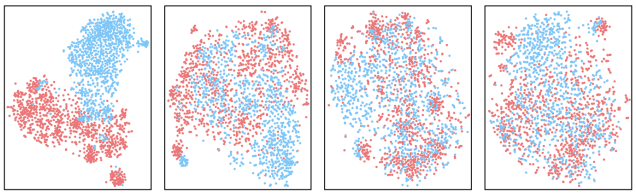


Figure 7. t-SNE visualization of real (red) and fake (blue) images using the feature space of different image encoders.

structure in separating the two classes, suggesting that the pre-training data matters more than the architecture.

6.4. Effect of training data source

So far, we have used ProGAN as the source of training data. We next repeat the evaluation setup in Table 2 using a pre-trained LDM [46] as the source instead. The real class consists of images from LAION dataset [48]. Fake images are generated using an LDM 200-step variant using text prompts from the corresponding real images. In total, the dataset consists of 400k real and 400k fake images.

Fig. 6 (top) compares our resulting linear classifier to the one created using ProGAN's dataset. Similar to what we have seen so far, access to only LDM's dataset also enables the model to achieve good generalizability. For example, our model can detect images from GAN's domain (now an unseen generative model), with an average of 97.32 mAP. In contrast, the trained deep network (Fig. 6 bottom) performs well only when the target model is from the same generative model family, and fails to generalize in detecting images from GAN variants, 60.17 mAP; i.e., the improvement made by our method for the unseen GAN domain is **+37.16 mAP**. In summary, with our linear classifier, one can start with ProGAN's data and detect LDM's fake images, or vice versa. This is encouraging because it tells us that, *so far*, with all the advancements in generative models, there is still a hidden link which connects various fake images.

7. Conclusion and Discussion

We studied the problem associated with training neural networks to detect fake images. The analysis paved the way for our simple fix to the problem: using an informative feature space *not trained* for real-vs-fake classification. Performing nearest neighbor / linear probing in this space results in a significantly better generalization ability of detecting fake images, particularly from newer methods like diffusion/autoregressive models. As mentioned in Sec. 6.4, these results indicate that even today there is something common between the fake images generated from a GAN and those from a diffusion model. However, what that similarity is remains an open question. And while having a better understanding of that question will be helpful in designing even better fake image detectors, we believe that the generalization benefits of our proposed solutions should warrant them as strong baselines in this line of work.

8. Acknowledgement

This work was supported in part by NSF CAREER IIS2150012, Institute of Information & communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. RS-2022-00187238, Development of Large Korean Language Model Technology for Efficient Pre-training), and Adobe Data Science Research Award.

References

[1] Adjust and exaggerate facial features. https://helpx.adobe.com/photoshop/how-to/face-awareliquify.html.

[2] Deepfacelab. https://github.com/iperov/deepfacelab.

[3] Dfaker. https://github.com/dfaker/df.

[4] Faceswap. https://faceswap.dev/.

[5] Shruti Agarwal and Hany Farid. Photo forensics from jpeg dimples. In *IEEE Workshop on Information Forensics and Security*, 2017.

[6] Vishal Asnani, Xi Yin, Tal Hassner, Sijia Liu, and Xiaoming Liu. Proactive image manipulation detection. In *CVPR*, 2022.

[7] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv*, 2018.

[8] Mu Cai and Yixuan Li. Out-of-distribution detection via frequency-regularized generative models. In *WACV*, 2023.

[9] Mu Cai, Hong Zhang, Huijuan Huang, Qichuan Geng, Yixuan Li, and Gao Huang. Frequency domain image translation: More photo-realistic, better identity-preserving. In *ICCV*, pages 13930–13940, 2021.

[10] Lucy Chai, David Bau, Ser-Nam Lim, and Phillip Isola. What makes fake images detectable? understanding properties that generalize. In *ECCV*, 2020.

[11] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *CVPR*, 2018.

[12] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, 2017.

[13] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, 2018.

[14] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *arXiv*, 2017.

[15] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Splicebuster: A new blind image splicing detector. In *IEEE International Workshop on Information Forensics and Security*, 2015.

[16] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva. Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection. In *arXiv*, 2017.

[17] Davide Cozzolino, Justus Thies, Rossler Andreas, Riess Christian, Nießner Matthias, and Luisa Verdoliva. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. In *arXiv*, 2019.

[18] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Zhang Lei. Second-order attention network for single image super-resolution. In *CVPR*, 2019.

[19] Cozzolino Davide, Thies Justus, Andreas Rossler, Matthias Nießner, and Luisa Verdoliva. Forensictransfer: Weakly-supervised domain adaptation for forgery detection. In *arXiv*, 2019.

[20] Boris Dayma, Suraj Patil, Pedro Cuenca, Khalid Saifullah, Tanishq Abraham, Phúc Lê Khac, Luke Melas, and Ritobrata Ghosh. Dall·e mini, 2021.

[21] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[22] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

[23] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

[24] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *arXiv*, 2020.

[25] Joel Frank, Thorsten Eisenhofer, Lea Schonherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *ICML*, 2020.

[26] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. In *IEEE Transactions on Information Forensics and Security*, 2012.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CVPR*, 2016.

[28] Terro Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*, 2018.

[29] Terro Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.

[30] Ke Li, Tianhao Zhang, and Jitendra Malik. Diverse image synthesis from semantic layouts via conditional imle. In *ICCV*, 2019.

[31] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. *CVPR*, 2023.

[32] Yuheng Li, Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. Mixnmatch: Multifactor disentanglement and encoding for conditional image generation. In *CVPR*, 2020.

[33] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 2008.

[34] Francesco Marra, Diego Gragnaniello, Cozzolino Davide, and Luisa and Verdoliva. Detection of gan-generated fake images over social networks. In *IEEE Conference on Multimedia Information Processing and Retrieval*, 2018.

[35] Lakshmanan Natraj, Tajuddin Manhar Mohammed, Shivkumar Chandrasekaran, Arjuna Flenner, Amit K. Roy-Chowdhuri, and B.S. Manjunath. Detecting gan generated fake images using co-occurrence matrices. In *Electronic imaging*, 2019.

[36] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022.

[37] James F. O'Brien and Hany Farid. Exposing photo manipulation with inconsistent reflections. In *ACM Transactions on Graphics*, 2012.

[38] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[39] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei A. Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. In *Advances in Neural Information Processing Systems*, 2020.

[40] Alin C. Popescu and Hany Farid. Exposing digital forgeries by detecting traces of resampling. In *IEEE Transactions on signal processing*, 2005.

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[42] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. In *arXiv*, 2022.

[43] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021.

[44] Yuan Rao and Jiangqun Ni. A deep learning approach to detection of splicing and copy-move forgeries in images. In *IEEE International Workshop on Information Forensics and Security*, 2016.

[45] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021.

[46] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[47] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner. FaceForensics++: Learning to detect manipulated facial images. In *International Conference on Computer Vision*, 2019.

[48] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. In *Data Centric AI NeurIPS Workshop 2021*, 2021.

[49] Sheng-Yu Wang, Oliver Wang, Andrew Owens, Richard Zhang, and Alexei A Efros. Detecting photoshopped faces by scripting photoshop. In *ICCV*, 2019.

[50] Sheng-Yu Wang, Oliver Wang, Richard Zhang, Andrew Owens, and Alexei A Efros. Cnn-generated images are surprisingly easy to spot...for now. In *CVPR*, 2020.

[51] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. 2015.

[52] Ning Yu, Larry S. Davis, and Mario Fritz. Attributing fake images to gans: Learning and analyzing gan fingerprints. In *ICCV*, 2019.

[53] Zu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *WIFS*, 2019.

[54] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017.

Fake-image detection with Robust Hashing

*

1st Miki Tanaka
Tokyo Metropoliltan University
Tokyo, Japan
tanaka-miki@ed.tmu.ac.jp

2nd Hitoshi Kiya
Tokyo Metropoliltan University
Tokyo, Japan
kiya@tmu.ac.jp

2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech) | 978-1-6654-1875-1/21/$31.00 ©2021 IEEE | DOI: 10.1109/LIFETECH52111.2021.9391842

***Abstract*—In this paper, we investigate whether robust hashing has a possibility to robustly detect fake-images even when multiple manipulation techniques such as JPEG compression are applied to images for the first time. In an experiment, the proposed fake detection with robust hashing is demonstrated to outperform state-of-the-art one under the use of various datasets including fake images generated with GANs.**

***Index Terms*—fake images, GAN**

I. Introduction

Recent rapid advances in image manipulation tools and deep image synthesis techniques, such as Generative Adversarial Networks (GANs) have easily generated fake images. In addition, with the spread of SNS (social networking services), the existence of fake images has become a major threat to the credibility of the international community. Accordingly, detecting manipulated images has become an urgent issue [1].

Most forgery detection methods assume that images are generated by using a specific manipulation technique, and the methods aim to detect unique features caused by the manipulation technique such as checkerboard artifacts [2]–[5]. Actually tampered images are usually uploaded to SNS and image sharing services. SNS providers are known to process the uploaded images by resizing or compressing them into JPEG format [6]–[9]. Such manipulation may damage or lose the unique features of fake images. However, the influence of manipulations on images has not been discussed sufficiently when a number of manipulation techniques such as JPEG compression are applied at the same time. In this paper, we investigate the possibility that there is a method with robust hashing that has been proposed for image retrieval, and the proposed method with robust hashing is demonstrated to have a high fake-detection accuracy, even when multiple manipulation techniques are carried out.

II. Related work

A. Fake-image generation

Fake images are manually generated by using image editing tools such as Photoshop. Splicing, copy-move, and deletion are also carried out under the use of such a tool. Similarly, resizing, rotating, blurring, and changing the color of an image can be manually carried out.

In addition, recent rapid advances in deep image synthesis techniques such as GANs have automatically generated fake images. CycleGAN [10] and StarGAN [11] are typical image synthesis techniques with GANs. CycleGAN is a GAN that performs one-to-one transformations, e.g. changing apples to oranges, while StarGAN is a GAN that performs many-to-many transformations, such as changing a person's facial expression or hair color (see Figs.1 and 3). Furthermore, fake videos created using deep learning are called Deepfake, and various tampering methods have emerged, such as those using autoencoders, Face2Face [12], FaceSwap [13], and so on.

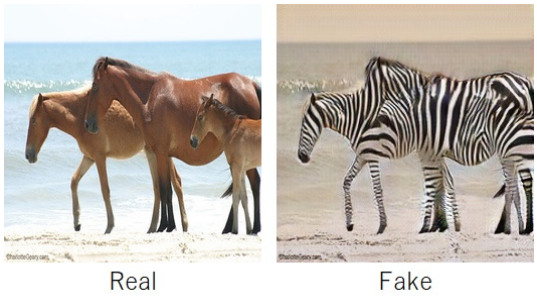


Fig. 1. Example Fake-images with CycleGAN

Real-world fake images may include the influence of a number of manipulation techniques such as image compression, resizing, copy-move at the same time, even if fake-images are generated by using GANs. Therefore, we have to consider such conditions for detecting real-world fake images.

B. Fake detection methods

Image tampering has a longer history than that of deep learning. Fragile watermarking [14], detection of double JPEG compression with a statistical method [15] [16], and use of PRNU (photo-response non-uniformity) patterns of each camera [17] [18] have been proposed to detect such tampers. However, most of them do not suppose to detect fake images generated with GANs. Moreover, they cannot detect the difference between fake images and just manipulated ones such as resized images, which are not fake images in general.

With the development of deep learning, fake detection methods with deep leaning have been studied so far. The methods with deep learning do not employ a reference image or the features of a reference image to detect tamper ones. The methods also assume that images are generated by using

978-1-6654-1875-1/21/$31.00 ©2021 IEEE

a specific manipulation technique to detect unique features caused by the manipulation technique.

There are several detection methods with deep learning for detecting fake images generated with an image editing tool as Photoshop. Some of them focus on detecting the boundary between tampered regions and an original image [19] [20] [21]. Besides, a detection method [22] enables us to train a model without tamper images.

Most detection methods with deep learning have been proposed to detect fake images generated by using GANs. An image classifier trained only with ProGAN was shown to be effective in detecting images generated by other GAN models [23]. Various studies have focused on detecting checkerboard artifacts caused in both of two processes: forward propagation of upsampling layers and backpropagation of convolutional layers [24]. In this work, the spectrum of images is used as an input image in order to capture the checkerboard artifacts.

To detect fake videos called DeepFake, a number of detection methods have been investigated so far. Some methods attempt to detect failures in the generation of fake videos, in terms of poorly generated eyes and teeth [25], the frequency of blinking as a feature [26], and the correctness of facial landmarks [27] or head posture [28]. However, all of these methods have been pointed out to have problems in the robustness against the difference between training datasets and test data [1]. In addition, the conventional methods have not considered the robustness against the combination of various manipulations such as the combination of resizing and DeepFake.

III. Proposed Method with Robust Hashing

A. Overview

Figure2 shows an overview of the proposed method. In the framework, robust hash value is computed from easy reference image by using a robust hash method, and stored in a database. Similar to reference images, a robust hash value is computed from a query one by using the same hash method. The hash value of the query is compared with those stored the database. Finally, the query image is judged whether it is real or fake in accordance with the distance between two hash values.

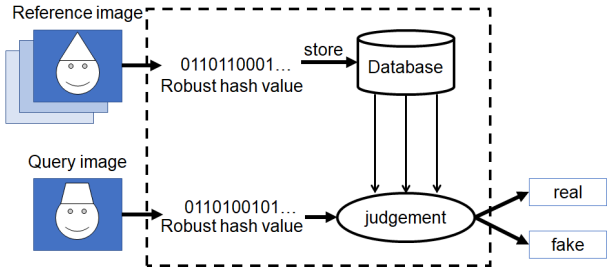


Fig. 2. Overview of proposed method

B. Fake detection with Robust Hashing

Various robust hashing methods have been proposed to retrieval similar images to a query one [29], [30]. In this paper, we apply the robust hashing method proposed by Li et al [29] for applying it to fake-image detection. This robust hashing enables us to robustly retrieve images, and has the following properties.

- Resizing images to 128×128 pixels prior to feature extraction.
- Performing 5×5-Gaussian low-pass filtering with a standard deviation of 1.
- Using rich features extracted from spatial and chromatic characteristics.
- Outputting a bit string with a length of 120 bits as a hash value.

In the method, the similarity is evaluated in accordance with the hamming distance between the hash string of a query image and that of each image in a database.

Let vectors $\boldsymbol{u} = \{u_1, u_2, \ldots, u_n\}$ and $\boldsymbol{q} = \{q_1, q_2, \ldots, q_n\}$, $u_i,\ q_i \in \{0, 1\}$ be the hash strings of reference image U and query image Q, respectively. The hamming distance $d_H(\boldsymbol{u}, \boldsymbol{q})$ between U and Q is given by:

$$d_H(\boldsymbol{u}, \boldsymbol{q}) \triangleq \sum_{i=1}^{n} \delta(u_i, q_i) \tag{1}$$

where

$$\delta(u_i, q_i) = \begin{cases} 0, & u_i = q_i \\ 1, & u_i \neq q_i \end{cases} . \tag{2}$$

To apply this similarity to fake-image detection, we introduce a threshold d as follows.

$$\begin{cases} Q \in \mathbb{U}', \min\limits_{u \neq q, u \in \mathbb{U}} (d_H(\boldsymbol{u}, \boldsymbol{q})) < d \\ Q \notin \mathbb{U}', \min\limits_{u \neq q, u \in \mathbb{U}} (d_H(\boldsymbol{u}, \boldsymbol{q})) \geq d \end{cases} \tag{3}$$

where $\mathbb{U}$ is a set of reference images and $\mathbb{U}'$ is the an of images generated with image manipulations from $\mathbb{U}$, which does not include fake images. According to eq. (3), Q is judged whether it is a fake image or not.

IV. Experiment Results

The proposed fake-image detection with robust hashing was experimentally evaluated in terms of accuracy and robustness against image manipulations.

A. Experiment setup

In the experiment, four fake-image datasets: Image Manipulation Dataset [31], UADFV [26], CycleGAN [10], and StarGAN [11] were used. The details of datasets are shown in Table I (see Figs. 1 and 3). The datasets consist of pairs of a fake-image and the original one. JPEG compression with a quantization parameter of $Q_J = 80$ was applied to all query images. $d = 3$ was selected as threshold d in accordance with the EER (Equal error rate) performance.

As one of the state-of-the-art fake detection methods, Wang's method [23] was compared with the proposed one. Wang's method was proposed for detecting images generated by using CNNs including various GAN models, where a classifier is trained by using ProGAN.

978-1-6654-1875-1/21/$31.00 ©2021 IEEE

TABLE I
DATASETS

dataset	Fake-image generation	real	fake
		No. of images	
Image Manipulation Dataset [31]	copy-move	48	48
UADFV [26]	face swap	49	49
CycleGAN [10]	GAN	1320	1320
StarGAN [11]	GAN	1999	1999

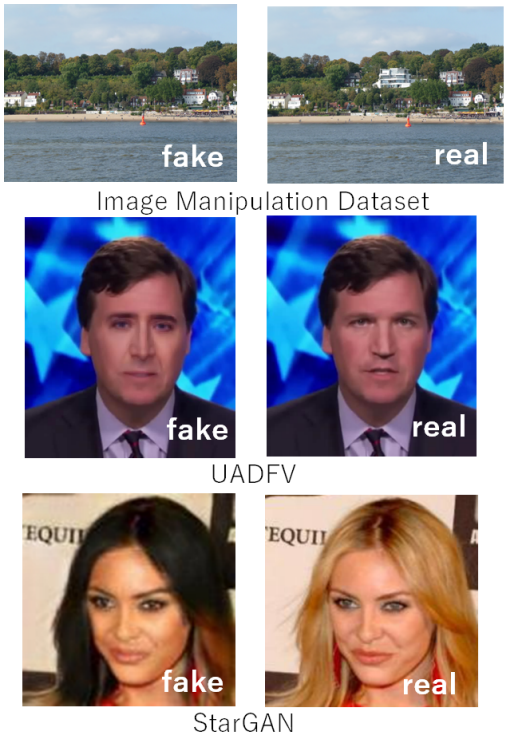


Fig. 3. Example of datasets

The performance of fake-image detection was evaluated by using AP (Average Precision) and Accuracy (fake), given by,

$$Accuracy\ (fake) = \frac{N_{tn}}{N_{Qf}} \tag{4}$$

where N_{Qf} is the number of fake query images, and N_{tn} is the number of fake query ones that are correctly judged as fake images.

B. Results without additional manipulation

Table II shows experimental results under the use of the two detection methods. From the table, it is shown that the proposed method had a higher performance than Wang's method in terms of both AP and Acc (fake). In addition, the performance of Wang's method heavily decreased when using the image manipulation and UADFV datasets. The reason is that Wang's method focuses on detecting fake images generated by using CNNs. The image manipulation dataset does not consist of images generated with GANs. In addition, although UADFV consists of images generated by using DeepFake, they have the influence of video compression.

TABLE II
COMPARISON WITH WANG'S METHOD

Dataset	Wang's method [23]		proposed	
	AP	Acc (fake)	AP	Acc (fake)
Image Manipulation Dataset	0.5185	0.0000	0.9760	0.8750
UADFV	0.5707	0.0000	0.8801	0.7083
CycleGAN	0.9768	0.5939	1.0000	1.0000
StarGAN	0.9594	0.5918	1.0000	1.0000

C. Results with additional manipulation

JPEG compression with $Q_J = 70$, resizing with a scale factor of 0.5, copy-move or splicing was applied to query images. Therefore, when query images were fake ones, the fake query ones included the effects of two manipulations at the same time.

Table III shows experimental results under the additional manipulation, where 50 fake images generated by using CycleGAN, in which horses were converted to zebras, were used (see Fig.1). The proposed method was confirmed to still maintain a high accuracy even under the additional manipulation. In contrast, Wang's method suffered from the influence of the addition manipulation. In particular, for splicing and resizing, Wang's method was affected by these operations. That is why the method assume that fake images are generated by using CNNs, to detect unique features caused by using CNNs. However, splicing and resizing don't depend on CNNs, although CycleGAN includes CNNs.

TABLE III
COMPARISON WITH WANG'S METHOD UNDER ADDITIONAL MANIPULATION (DATASET: CYCLEGAN)

additional manipulation	Wang's method [23]		proposed	
	AP	Acc (fake)	AP	Acc (fake)
None	0.9833	0.6200	0.9941	1.0000
JPEG($Q_J = 70$)	0.9670	0.6000	0.9922	0.9800
resize (0.5)	0.8264	0.2400	0.9793	1.0000
copy-move	0.9781	0.6000	1.0000	1.0000
splicing	0.9666	0.4800	0.9992	1.0000

V. CONCLUSION

In this paper, we proposed a novel fake-image detection method with robust hashing for the first time. Although various robust hashing methods have been proposed to retrieve similar images to a query one so far, a robust hashing method proposed by Li et al was applied to various datasets including fake images generated with GANs. In the experiment, the proposed method was demonstrated not only to outperform a state-of-the-art but also to be robust against the combination of image manipulations.

978-1-6654-1875-1/21/$31.00 ©2021 IEEE

REFERENCES

[1] L. Verdoliva, "Media forensics and deepfakes: An overview," *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 910–932, 2020.

[2] Y. Sugawara, S. Shiota, and H. Kiya, "Super-resolution using convolutional neural networks without any checkerboard artifacts," in *Proc. of IEEE International Conference on Image Processing*, 2018, pp. 66–70.

[3] Y. Sugawara, S. Shiota, and H. Kiya, "Checkerboard artifacts free convolutional neural networks," *APSIPA Transactions on Signal and Information Processing*, vol. 8, p. e9, 2019.

[4] Y. Kinoshita and H. Kiya, "Fixed smooth convolutional layer for avoiding checkerboard artifacts in cnns," in *Proc. in IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 3712–3716.

[5] T. Osakabe, M. Tanaka, Y. Kinoshita, and H. Kiya, "Cyclegan without checkerboard artifacts for counter-forensics of fake-image detection," *arXive preprint arXive:2012.00287*, 2020. [Online]. Available: https://arxiv.org/abs/2012.00287

[6] T. Chuman, K. Iida, W. Sirichotedumrong, and H. Kiya, "Image manipulation specifications on social networking services for encryption-then-compression systems," *IEICE Transactions on Information and Systems*, vol. E102.D, no. 1, pp. 11–18, 2019.

[7] T. Chuman, K. Kurihara, and H. Kiya, "Security evaluation for block scrambling-based etc systems against extended jigsaw puzzle solver attacks," in *Proc. of IEEE International Conference on Multimedia and Expo (ICME)*, 2017, pp. 229–234.

[8] W. Sirichotedumrong and H. Kiya, "Grayscale-based block scrambling image encryption using ycbcr color space for encryption-then-compression systems," *APSIPA Transactions on Signal and Information Processing*, vol. 8, p. e7, 2019.

[9] T. Chuman, W. Sirichotedumrong, and H. Kiya, "Encryption-then-compression systems using grayscale-based image encryption for jpeg images," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1515–1525, 2019.

[10] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. of IEEE International Conference on Computer Vision*, Oct 2017.

[11] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "Stargan: Unified generative adversarial networks for multi-domain image-to-image translation," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, June 2018.

[12] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Niessner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, June 2016.

[13] Y. Nirkin, I. Masi, A. Tran Tuan, T. Hassner, and G. Medioni, "On face segmentation, face swapping, and face perception," in *Proc. of IEEE International Conference on Automatic Face Gesture Recognition*, 2018, pp. 98–105.

[14] A. T. S. Ho, X. Zhu, J. Shen, and P. Marziliano, "Fragile watermarking based on encoding of the zeroes of the z-transform," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 567–569, 2008.

[15] G. Zhenzhen, N. Shaozhang, and H. Hongli, "Tamper detection method for clipped double jpeg compression image," in *Proc. of International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2015, pp. 185–188.

[16] T. Bianchi and A. Piva, "Detection of nonaligned double jpeg compression based on integer periodicity maps," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 842–848, 2012.

[17] M. Chen, J. Fridrich, M. Goljan, and J. Lukas, "Determining image origin and integrity using sensor noise," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 1, pp. 74–90, 2008.

[18] G. Chierchia, G. Poggi, C. Sansone, and L. Verdoliva, "A bayesian-mrf approach for prnu-based image forgery detection," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 4, pp. 554–567, 2014.

[19] Y. Rao and J. Ni, "A deep learning approach to detection of splicing and copy-move forgeries in images," in *Pros. of IEEE International Workshop on Information Forensics and Security*, 2016, pp. 1–6.

[20] J. H. Bappy, A. K. Roy-Chowdhury, J. Bunk, L. Nataraj, and B. S. Manjunath, "Exploiting spatial structure for localizing manipulated image regions," in *Proc. of IEEE International Conference on Computer Vision*, Oct 2017.

[21] P. Zhou, X. Han, V. I. Morariu, and L. S. Davis, "Pros. of learning rich features for image manipulation detection," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, June 2018.

[22] M. Huh, A. Liu, A. Owens, and A. A. Efros, "Pros. of fighting fake news: Image splice detection via learned self-consistency," in *Proc. of European Conference on Computer Vision*, September 2018.

[23] S.-Y. Wang, O. Wang, R. Zhang, A. Owens, and A. A. Efros, "Cnn-generated images are surprisingly easy to spot... for now," in *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020.

[24] X. Zhang, S. Karaman, and S. Chang, "Detecting and simulating artifacts in gan fake images," in *Proc. of IEEE International Workshop on Information Forensics and Security*, 2019, pp. 1–6.

[25] F. Matern, C. Riess, and M. Stamminger, "Exploiting visual artifacts to expose deepfakes and face manipulations," in *Proc. of IEEE Winter Applications of Computer Vision Workshops*, 2019, pp. 83–92.

[26] Y. Li, M. Chang, and S. Lyu, "In ictu oculi: Exposing ai created fake videos by detecting eye blinking," in *Proc. of IEEE International Workshop on Information Forensics and Security*, 2018, pp. 1–7.

[27] X. Yang, Y. Li, H. Qi, and S. Lyu, "Exposing gan-synthesized faces using landmark locations," in *Proc. of ACM Workshop on Information Hiding and Multimedia Security*, 2019, p. 113–118.

[28] X. Yang, Y. Li, and S. Lyu, "Exposing deep fakes using inconsistent head poses," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019, pp. 8261–8265.

[29] Y. N. Li, P. Wang, and Y. T. Su, "Robust image hashing based on selective quaternion invariance," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2396–2400, 2015.

[30] K. Iida and H. Kiya, "Robust image identification with dc coefficients for double-compressed jpeg images," *IEICE Transactions on Information and Systems*, vol. E102.D, no. 1, pp. 2–10, 2019.

[31] "Image manipulation dataset," https://www5.cs.fau.de/research/data/image-manipulation/.

978-1-6654-1875-1/21/$31.00 ©2021 IEEE