Open Set Synthetic Image Source Attribution

Shengbang Fang
sf683@drexel.edu
Tai D. Nguyen
tdn47@drexel.edu
Matthew C. Stamm
mcs382@drexel.edu

ECE Department
Drexel University
Philadelphia, PA, USA

Abstract

AI-generated images have become increasingly realistic and have garnered significant public attention. While synthetic images are intriguing due to their realism, they also pose an important misinformation threat. To address this new threat, researchers have developed multiple algorithms to detect synthetic images and identify their source generators. However, most existing source attribution techniques are designed to operate in a closed-set scenario, i.e. they can only be used to discriminate between known image generators. By contrast, new image generation techniques are rapidly emerging. To contend with this, there is a great need for open set source attribution techniques that can identify when synthetic images have originated from new, unseen generators. To address this problem, we propose a new metric learning-based approach. Our technique works by learning transferrable embeddings capable of discriminating between generators, even when they are not seen during training. An image is first assigned to a candidate generator, then is accepted or rejected based on its distance in the embedding space from known generators' learned reference points. Importantly, we identify that initializing our source attribution embedding network by pretraining it on image camera identification can improve our embeddings' transferability. Through a series of experiments, we demonstrate our approach's ability to attribute the source of synthetic images in open-set scenarios.

1 Introduction

As deep learning techniques have evolved rapidly in recent years, AI-based image synthesis algorithms have become increasingly successful and ubiquitous. Researchers have used techniques such as variational autoencoders [27, 38], GAN based generators [15], diffusion models [18, 19, 38], and other techniques [13, 30, 35, 50] to generate images that look like they were captured by a real camera. While some argue that these techniques enable artists be more creative, synthetic images can also be employed for malicious purposes, including online misinformation and disinformation campaigns.

In order to address the potential threats posed by novel AI-generated multimedia content, researchers have developed a wide variety of highly-performing algorithms aimed at detecting synthetic images. Wang et al. [46] analyzed the periodic signal present in the frequency

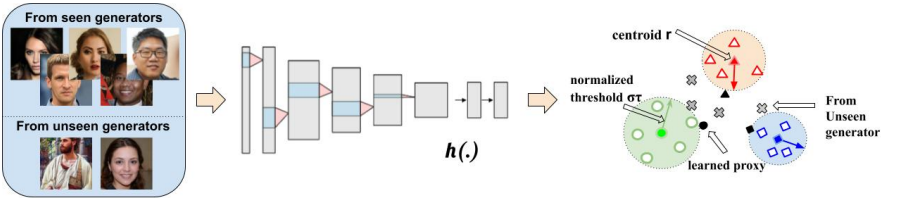

Figure 1: Proposed method's pipeline. We proposed a metric-learning based approach for open set synthetic image source attribution. Where images from the seen generators by the algorithm will be accurately classified to be their original generator, and images from unseen generators will be reject by the normalized threshold.

domain of synthetic images, which is caused by up-sampling in the neural network generator. Other methods have also been proposed for detecting synthetic images using learned forensic traces [39].

However, with the widespread use of synthetic images, it has become increasingly important to trace their sources. That is because the images' origins can be used to trace and reveal the nature of a particular misinformation or disinformation campaign. For example, it is important to know if a synthetic image circulated online came from a known source or from a new, unknown origin To achieve this, researchers have developed several approaches to attribute the source of synthetic images. Frank et al. [11] proposed using DCT-CNN to discriminate generators from DCT-transformed images. Bui et al. [5] proposed a feature mix-up method to train a synthetic image source attribution model. Albright et al. [1] proposed an inverse method to identify if an image belongs to the target generator by training proxy models for each of them.

While these approaches have great reported performance, they are only designed to operate in a closed-set classification scenario and cannot be easily adapted to work on new sources. This is problematic because new generators and generation techniques are rapidly emerging. Therefore, closed-set synthetic source attribution algorithms are not suitable to be used in the real world. And, to the best of our knowledge, comparatively less work had been proposed to solve this urgent and novel problem. Girish et al. [12] proposed an iterative clustering algorithm to cluster a large amount of images from unseen generators into groups. However, this work focuses on grouping a large set of synthetic images into many self-similar clusters (typically many more clusters than there are potential sources). It does not provide an identification criteria to predict whether an image came from a known generator or an unknown one.

In this paper, we propose a new algorithm based on metric learning to perform open set synthetic image source attribution. Our method involves learning transferable embeddings capable of differentiating between synthetic image generators, including those never encountered during training. We achieve this by identifying reference points in the embedding space associated with each known generator. To attribute a synthetic image to its source, we first determine the nearest candidate class reference to the query image's embedding. We then use the distance between the query image's embedding and the identified class reference to either accept or reject the candidate source. If the distance is below a threshold, we accept the candidate source, otherwise, we identify the image to be from an unknown source generator. We summarize our contributions as follows:

- We develop a new algorithm that can perform open-set synthetic image attribution and

outperforms existing approaches.

- We develop a new metric-learning based embedding to measure the similarity between synthetic images' source generators.
- We propose a new reject criteria to determine if a query image is from a new generator.
- We demonstrate that pretraining the embedding network to perform camera model classification can improve model transferability on images from unseen generators.

2 Related Work

Image Synthesis Algorithms Researchers proposed multiple image generation techniques, including Variational Auto Encoder [27], GAN [15] and the recent diffusion-based models [18, 38]. These image synthesis techiques have been widely used to generate different image contents, including prompt-guided image synthesis [29, 38], and super-resolution [19].

Forensic Algorithms To combat falsified content in images, researchers in the forensic community developed signal processing based methods for forgery detection. These algorithms are often based on human-designed features to detect the inconsistencies in forensic traces in the frequency domain [12, 26, 28, 28, 36, 37, 41, 42]. Additionally, researchers also developed multiple deep-learning based forensic algorithms to detect both traditional image editing [2, 3], and AI-generated contents [39]. In particular, [3, 47] developed CNNs with high-pass filters to extract generic forensic features from images. This had been proven to be effective at detecting image forgery.

Synthetic Image Detection Researchers have created multiple algorithms to detect traces left by a generator or generator's architecture in synthetic images [45]. Previous research [46] showed that images from different generators architectures, including deepfake ones, often contain distinctive high-frequency information. Additionally, other techniques like inversion [1] is also developed for synthetic image detection. This technique works by inverting the generator to obtain the set of features that were used to generate the synthesized image. The authors showed that these features can be used to attribute specific generators.

Synthetic Image Source Attribution Researchers have developed many algorithms for synthetic image attribution [1, 5, 49]. DCT-CNN [11] utilized the high-frequency information by first converting the input image using discrete cosine transform, then used a shallow CNN to perform closed-set source attribution. [5] proposed a new feature mix-up method, and trained the model using a well-balanced training data to perform source attribution in a closed-set scenario. However, currently research on open-set image source attribution remains limited. Sharath el. [14] proposed a method to cluster large amount of images from unseen generators into groups. This is different from source attribution on a single query image, because it does not have an identification mechanism to predict if an image came from a known or unknown generator. To the best of our knowledge, there is no existing algorithm can perform open-set synthetic image source attribution.

Other Open-set Image Source Attribution Forensic researchers developed many open-set image camera model attribution algorithms [4, 9, 20, 21, 32]. These algorithms often leveraged the Siamese network architecture, in which a pair of images from the same camera model is labeled similar, while a pair of images from different camera models is labeled different. These algorithms have been proven to be efficient and transferable to multiple different types of image forgery and manipulation.

3 Proposed Method

In this section, we introduce our proposed approach for open-set synthetic image source attribution. The problem we target at is: Given several known synthetic image generators, for a synthetic image, the algorithm must accomplish two tasks: 1. Determine whether or not such image comes from a known generator; and 2. Attribute the source if it is from a known generator, or, predict the image to be from some unseen source if it is from an unknown generator.

For a more accurate definition, assume the investigator has a set of known image generators $\mathbb{G}$ and a set of synthetic images $\mathbb{T}$ created by $\mathbb{G}$. $\mathbb{T}$ is partitioned into disjoint subsets $\mathbb{T}_i$, each belongs to a generator architecture $g_i \in \mathbb{G}$. For a good open-set source attribution approach, the algorithm should correctly identify the source of an synthetic image as g_i if it is from a known generator $g_i \in \mathbb{G}$, or attribute the source to be unknown if the origin is an unknown generator $g_u \notin \mathbb{G}$. Overall, an open-set algorithm requires a source identification rule $S : \mathbb{X} \to \mathbb{G}$, which assigns a candidate generator within the known set $\mathbb{G}$, and a rejection rule $R : \mathbb{X} \times \mathbb{G} \to \{0, 1\}$ where 0 indicates that the class identified by S should be rejected and the image comes from an unknown generator, and 1 indicates that the candidate class should be accepted.

While existing closed-set approaches are good at producing S, they are not designed to produce R. To address this problem, we propose a source identification rule S defined by an embedding function $h(\cdot)$, a distance metric $d(\cdot, \cdot)$, and a set of class references r_k on the embedding space. For a synthetic image x, the source identification rule S is made by measuring the distance between its embedding $h(x)$ and the reference points r_k for each class in the embedding space using $d(\cdot, \cdot)$. The generator associated to the closest reference point to $h(x)$ is identified as the candidate source $g \in \mathbb{G}$.

It is critical that h is generic and can capture forensic traces that can both accurately discriminate images among known sources and identify images from unknown sources. To achieve this, we propose a novel training procedure in which we initialize h by pretraining the network to perform camera model classification using the Camera Model Database [32, 33, 34]. Previous studies have shown that this process enables the neural network to learn generic embeddings that can be transferred to other forensic tasks [34]. Our experiments demonstrate that this procedure significantly improves the model's open-set performance.

After making the closed-set source identification decision with the rule S, we determine whether to accept or reject the source generator candidate using rule R. This is achieved by first normalizing the distance between the query image's embedding $h(x)$ and the reference point r_i based on the chosen generator g_i. Then, if the normalized distance exceeds a threshold τ, we reject the source decision and consider it to be from an unknown generator. Otherwise, we accept and identify it as being from source g_i.

3.1 Transferable Embedding Initialization

It is crucial that the embedding function h both accurately discriminate between images from known sources and identify those from unknown sources. A common way to learn an embedding is to train a CNN to discriminate between generators in the known set. However, this does not necessarily learn an embedding that transfer well to unknown generators. To address this issue, we propose novel training procedure, in which we first initialize h by pretraining on the task of camera model identification with large number of classes. By doing so, we enable h to be more sensitive to small differences in forensic traces, and make

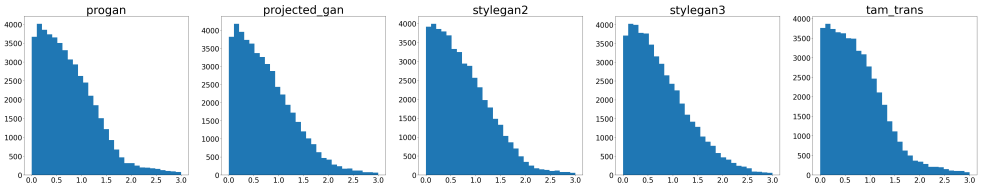

Figure 2: Distribution of training set $\mathbb{T}$'s embeddings' normalized distance from reference point. We choose Xception to be the h. This figure shows the normalized distance is a good metric for rule R in proposed approach.

h's embedding more transferable to images of unknown sources. Our experiments showed that this procedure significantly improves the open-set performance.

$$\mathcal{L}_{init} = -\sum_{k} y_k log(\hat{y}_k) \tag{1}$$

After pretraining, we removed the last classification layer and used the feature maps before the final layer as the initialization for h. Following this procedure, we fine-tuned h to learn a more precise embedding that could effectively discriminate between different image generators in the following section.

3.2 Embedding Learning

Metric learning algorithms enable a model to adapt to unseen types of data during the training process. These techniques learn an embedding and associated metric such that data points of the same class are closed together and data points of different classes are far apart. By utilizing this characteristics, we can identify images from unknown generators by testing if their embeddings lie far away from known generators' embedding clusters. To accomplish this, we utilize ProxyNCA++ [43] to further train our initialized embedding function h. ProxyNCA++ has been widely used to learn embeddings for open set tasks, and has been shown to have higher accuracy and faster convergence compared to other non-proxy based methods. During training, it uses randomly initialized proxies for each class, pull the embeddings closer to their assigned proxy and push them away from other classes' proxies. The ProxyNCA++ loss can be written as:

$$P_i = \frac{exp(-d(h(x_i), p(y_i)))}{\sum_{p(a)\in p(\mathcal{A})} exp(-d(h(x_i), p(a)))} \tag{2}$$

$$\mathcal{L}_{proxyNCA++} = -log(P_i) \tag{3}$$

where y_i is the generator architecture label of x_i, $p(y_i)$ is the corresponding proxy embedding of label y_i. The x is the image, h is the feature extractor. Set $\mathcal{A}$ contains all generator classes in the training set. For the distance metric $d(\cdot,\cdot)$, we choose L_2 distance to be the metric.

3.3 Finding Class References

After completing the metric learning process and obtaining h, we compute the centroid embedding r_i for each class i in the learned embedding space, using all images from generator g_i in the training data $T_i \in \mathbb{T}$. We compute the centroid of each class' embeddings over the training set as its reference point:

$$r_i = \frac{\sum_{x_i \in \mathbb{T}_i} h(x_i)}{|\mathbb{T}_i|} \tag{4}$$

To perform inference, we build S by choosing x's nearest reference point r_i to assign a potential generator class i:

$$S(x) = \underset{k}{\operatorname{argmin}}\, d(h(x), r_k) \tag{5}$$

where $d(h(x), r_k)$ is the L_2-distance between x's embedding with the reference point r_k of class k. After we assign an class i, we use a reject criteria to test if x belongs to the class i or is from unseen generator.

3.4 Reject Criteria

For open set source attribution, we establish a rejection criterion R for all seen classes. We use the normalized distance between $h(x)$ and r_i to determine whether to accept or reject the class decision i. This is because each class' embedding group may have different variance, and normalization of the distance enables a better measurement of the relative similarity for each class. The normalized distance $s(x, r_i)$ between $h(x)$ and r_i is defined as:

$$s(x, r_i) = \frac{d(h(x), r_i)}{\sigma_i} \tag{6}$$

where σ is computed inspired by sample standard deviation of Gaussian distribution:

$$\sigma_i = \sqrt{E(d(h(x_i), r_i)^2)} = \sqrt{E(||h(x_i) - r_i||_2^2)} = \sqrt{\frac{\sum_{x_i \in \mathbb{T}_i} ||h(x_i) - r_i||_2^2}{|\mathbb{T}_i| - 1}} \tag{7}$$

This normalization enable R to adapt to variance in the embedding space for each class. Fig 2 show san example of the distribution of normalized distances over each $\mathbb{T}_i$ in an embedding space, using Xception [8] as the embedding network h. This figure illustrates that normalization can map the distance between $x \in g_i$ and different r_i into a similar distribution.

After that, we define the rejection criterion by comparing the normalized distance $s(x, r_i)$ between $h(x)$ and r_i with a threshold τ. The threshold τ is applied to all seen classes of the generator. If $s(x, r_i) < \tau_i$, we accept x_i and classify it as coming from a seen generator g_i. Otherwise, we reject x and identify it as coming from a new unknown image generator g_u:

$$R(x) = \begin{cases} g_i \in \mathbb{G}, & \text{if } s(x, r_i) < \tau \\ g_u \notin \mathbb{G}, & \text{if } s(x, r_i) \geqslant \tau \end{cases} \tag{8}$$

4 Experimental Results

This section demonstrates our approach for open-set synthetic image attribution through multiple experiments. We first describe the dataset we create to train and benchmark our proposed algorithm, then we introduce the hyper-parameters we use to train the embedding function h. We also propose and discuss the metrics we use for evaluating algorithms' performance. Then, we display our proposed approach's performance on open-set synthetic image attribution. Lastly, we compare our best model with state-of-the-art publicly available synthetic image attribution approaches.

Generator	Dataset for Generator Training
ProGAN	celebA, lsun-churchoutdoor, lsun-bicycle, lsun-bird, lsun-bedroom, lsun-car
ProjectedGAN	ffhq, lsun-bedroom, lsun-churchoutdoor, cityscape
Tam-Transformer	ffhq, imagenet-fish, imagenet-shark, imagenet-dog
StyleGAN2	ffhq, metfaces, afhqv2, afhqdog
StyleGAN3	ffhq, metfaces, afhqv2
StyleGAN	ffhq, celebahq, lsun-bedroom, lsun-car
Stable Diffusion	N/A

Table 1: Generator architectures, and corresponding datasets that are used to train the generator for image synthesizing

The results show that our approach can successfully attribute the source generator's architecture of an image in an open-set scenario. Additionaly, we find that other existing closed-set method cannot be used to analyze images from unseen models, and other existing open-set forensic algorithms do not perform well on synthetic image attribution. Furthermore, we show that pre-training our models on image camera model classification improves there generalizability in attributing synthetic images

4.1 Dataset

We created our own dataset to train the proposed method. We used seven publicly available image generators and synthesize multiple images from each. The seven generators are ProGAN [22], Projected-GAN [40], StyleGAN [23], StyleGAN2 [24], StyleGAN3 [25], Taming Transformer [10], and Stable Diffusion [38]. The dataset we used for training the generator are listed in Table.1. For stable diffusion, we directly used the publicly available model.

To create our dataset, we generated 10,000 synthetic images for each combination of generator and dataset. Then, for each combination, we picked 8,000 images for training, 1,000 images for validation, and 1,000 images for testing. We randomly picked StyleGAN and Stable Diffusion and hold out the images from them during training, and used them as the "unseen" group of generator architecture for evaluating model's performance on the open-set attribution task. We observe that the performance of different combination of held-out set have comparable performance with statistical deviation. Since synthetic images are often undergo JPEG compression in the real world, we randomly JPEG-compressed images using different quality factors of 75, 80, 85, 90, 95, along with no compression with equal probability. During training and evaluation, when the image size mismatches with the model's input size, we apply random cropping to 256 by 256 pixels. We intentionally avoid resampling to preserve the forensic features and information of the synthetic images.

4.2 Training Parameters

We trained our model using two training protocols to demonstrate the benefits of pre-training on the camera identification task. The first involved training the model from initial weights that were obtained through pre-training, and the second involved training the model from scratch. During all training stages, we apply under-sampling over all seen generators to balance the number of images for each class.

Train From Pre-training Initialization We chose 70 camera models from the Camera Model Identification Database used in [32, 33, 34] for classification. We used AdamW [31] optimizer, with an initial learning rate of 0.001, decayed with a scale of 0.65 every 3 epochs. After training, we discarded the classification layer and trained the rest of the model with

Embedding Arch.	Xception	ResNet50	CamID-CNN	Stega-CNN	MISLNet
Train From Scratch	0.827	0.671	0.519	0.787	0.761
With Pre-Training	**0.868**	0.714	0.574	0.808	**0.868**

Table 2: AUC of $aF_1 - CRR$ response curve for different models. Pre-training on camera model classification significantly improved the models' generalizability and performance on open-set identification scenario.

ProxyNCA++ loss for 30 epochs. The initial learning rate for this stage is 0.0001, which decayed with a scale of 0.6 every 2 epochs.

Train From Scratch We trained our embedding models from scratch for 30 epochs, still using AdamW [31] optimization with an initial learning rate of 7e-4, decay for 0.6 for every 2 epochs. We chose the model with a highest validation accuracy on "seen" generator.

4.3 Evaluation Metrics

To evaluate different approaches for open-set synthetic image attribution, we used the average F-1 scores (aF_1) on seen generator architectures and the correct reject rate (CRR) on unseen architectures. The aF_1 estimates how well the generators perform on seen generator architectures under a specific threshold, balancing the recall and precision for detecting each generator's images. The aF_1 score is defined as the F-1 score averaged over 5 seen $g_s \in \mathbb{G}$:

$$aF_1 = \sum_{j=1}^{N} F_{1j}/N, \quad where \;\; F_{1j} = \frac{2TP_j}{2TP_j + FP_j + FN_j} \tag{9}$$

where TP_j is the number of samples from generator architecture j being predicted from generator j. FP_j is the number of samples not from generator architecture j, including both seen and unseen architectures, to be predicted from generator j. FN_j is the number of samples from generator j but are not classifier from j. N is the number of seen generators.

The correct reject rate (CRR) measures the ability of the model to reject an image to be from an unseen generator architectures $g_u \notin \mathbb{G}$. It is defined as:

$$CRR = \frac{|\,\{\min_i s(x, r_i) > \tau, \;\; \forall x \in \{g_u \notin \mathbb{G}\}\}\,|}{|\,\{\forall x \in \{g_u \notin \{\mathbb{G}\}\}\,|} \tag{10}$$

which is the probability of images from unseen generator architecture g_u (StyleGAN, Stable Diffusion) being rejected. The higher the CRR, the better the discrimination ability of the embedding function h.

4.4 Choosing Optimal Embedding Function

We conduct a series of experiments to choose the best embedding function h. For the embedding architecture of h, we evaluated five different CNNs and compared their performance to determine the best candidate. The five CNNs are Xception [8], ResNet50 [17], CamID CNN [44], Stega-CNN [48] and MISLNet [3]. Among them, Xception and ResNet50 were originally designed to perform object recognition tasks, both of them have also widely used by the forensic community on image forensic tasks [5, 6, 7, 16, 20, 46]. Stega-CNN was designed for steganalysis tasks. CamID CNN and MISLNet were originally designed for camera model identification. We choose these architectures because previous research has shown that they performed well in learning low-level forensic features from images.

Benefit Of Pre-Training On Camera Model Identification We conducted an experiment to verify the importance of pre-training our embedding architecture before applying metric learning. We did this by evaluating the AUC of the $aF_1 - CRR$ curve for all five embedding architectures with and without camera model identification pre-training.

From the result displayed in Table 2, we see that the pre-training on camera model improved all five model's AUC of $aF_1 - CRR$ curve. Among them, Xception and MISLNet achieved the best performance with an AUC of 0.868. This result demonstrates that pre-training on the camera model classification provides useful prior information to learn synthetic image embeddings and improves the model's transferability in an open-set scenario.

Model Selection We compared the performance of each embedding architecture with pretraining. From Table 2, we can see that both MISLNet and Xception achieve the highest performance. In the following experiments, we select MISLNet as the embedding function h for further evaluation. We choose MISLNet because it is a very light-weight architecture and can potentially have less over-fitting during training.

4.5 Open Set Attribution Performance

Method	ProGAN	Proj.-GAN	StyleGAN2	StyleGAN3	Taming Trans.	aF_1	StyleGAN	Stable Diffusion	CRR
RepMix	0.669	0.827	0.762	0.839	0.860	0.791	0	0	0
DCT-CNN	0.673	0.929	0.687	0.609	0.851	0.750	0	0	0
ResNet-50	0.572	0.995	0.995	0.797	0.976	0.867	0	0	0
Proposed	0.744	0.974	0.875	0.969	0.940	**0.900**	0.484	0.806	**0.645**
FSM	0.000	0.032	0.000	0.385	0.585	0.200	0.910	0.363	0.637
EXIF-Net	0.374	0.245	0.124	0.187	0.163	0.219	0.525	0.741	0.633

Table 3: Comparison with other existing approaches on open-set synthetic image attribution. Competing algorithms include: closed-set synthetic image attribution approaches, and open-set image source identification approaches. Results shows that our proposed outperformed both types of algorithms.

We evaluate our proposed approach's ability to perform open set source attribution using our final selection (MISLNet) for the embedding architecture. We compared our performance to several existing closed set synthetic image source attribution approaches, namely DCT-CNN [11] and RepMix [5]. Additionally, we trained ResNet-50 to perform synthetic image source attribution, as this network has been widely used in other publications to perform synthetic image detection [5, 20, 46]. We note that to the best of our knowledge, there are no directly comparable open-set source synthetic image source attribution approaches. We trained these CNNs as classifiers with training set images from 5 known generators.

Furthermore, we compared our performance to several open-set approaches to measure the similarity between forensic traces, namely FSM [32] and EXIFNet [20]. While not directly trained to perform synthetic image source ID, these networks are designed to determine the similarity of forensic traces between two image patches. We used the publicly available implementations of these algorithms to perform synthetic image source attribution by applying the same open-set identification mechanism with our approach. To compute the reference point r_i for each class i, we computed the average embedding output by the Siamese arm's feature extractor. For $d(\cdot,\cdot)$ we directly used the similarity metric on top of the two siamese arms. The evaluation result is shown in Table 3. From the results we see that our proposed approach can successfully identify images from known generators and reject images from unknown generators. Our approach also outperforms existing algorithms.

Comparison With Closed Set Approaches Table 3 shows the resulting comparison of our approach and other existing closed-set approaches (RepMix, DCT-CNN, and ResNet-50). We achieved higher aF_1 on the set of seen generator architectures than competing closed-set approaches. Furthermore, our approach obtained an average CRR of 0.645 while these methods can only get a CRR of 0. This shows that in open set scenarios, we are able to outperform these techniques in terms of both aF_1 and CRR. We note that we can have even higher aF_1 score at the cost of lowered CRR.

Comparison With Open Set Forensic Algorithms From Table 3, we see that our proposed method obtained a higher CRR than existing open-set approaches. Furthermore, we achieved a substantially higher aF_1 on the set of known sources than these networks. This demonstrates that while existing open-set approaches are able to reject images to be from an unknown generator architecture, they have very little ability to correctly identify known sources. Hence, existing open-set forensic algorithms cannot adapt to the synthetic images attribution task. However, our proposed method can both reliably identify if an image comes from a generator architecture that was seen during training, and reject an image coming from an unknown architecture.

4.6 Why Camera Identification Pre-Training Helps

Here we discuss some potential reason why forensic features learned when performing camera model classification are transferable to attributing synthetic images. Previous research [46, 51] identified distinct high-frequency patterns left by different generator architectures in the frequency domain. These patterns are largely due to the specific up-sampling operations utilized in these networks when generating a final resolution image. We note that these patterns are very similar to the high-frequency pattern of image-resampling, JPEG compression, and double-JPEG compression. In practice, different camera models contain different imaging sensors, different noise pattern, and use different JPEG-compression settings. All these processes leave unique high-frequency traces, which enable reliable camera model classification. Therefore, training a network to discriminate between different camera models allows the network to learn these high frequency traces. Pre-training an embedding architecture to perform camera model identification results in more transferability to the synthetic image attribution task because synthetic image generators can be discriminated on the basis of similar high-frequency traces.

5 Conclusion

In this paper, we proposed a new algorithm to perform open-set synthetic image attribution. Through extensive experiments, we demonstrate that our system can successfully perform open-set synthetic image attribution, and outperforms existing methods. In our approach, we use metric-learning to learn an embedding, and compare synthetic images' source generators by distance measuring its embedding's distance from a class reference in the embedding space. We also propose a new accept/reject criteria for images from unseen generators in open-set scenario. Additionally, we demonstrate that pre-training an embedding network to perform camera model identification helps improve its transferrability to unknown generators when performing synthetic image attribution. Through a set of experiments, we verify the importance of embedding pre-training and show that our proposed approach can succesfully perform open set synthetic image source attribution.

Acknowledgment

This work was supported in part by the Army Research Office and was accomplished under Cooperative Grant W911NF-20-2-0111 and by DARPA and Air Force Research Laboratory (AFRL) under agreement number HR0011-20-C-0126.

References

[1] Michael Albright and Scott McCloskey. Source generator attribution via inversion. In *CVPR Workshops*, volume 8, 2019.

[2] Belhassen Bayar and Matthew C Stamm. A deep learning approach to universal image manipulation detection using a new convolutional layer. In *Proceedings of the 4th ACM workshop on information hiding and multimedia security*, pages 5–10, 2016.

[3] Belhassen Bayar and Matthew C Stamm. Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection. *IEEE Transactions on Information Forensics and Security*, 13(11):2691–2706, 2018.

[4] Belhassen Bayar and Matthew C Stamm. Towards open set camera model identification using a deep learning framework. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 2007–2011. IEEE, 2018.

[5] Tu Bui, Ning Yu, and John Collomosse. Repmix: Representation mixing for robust attribution of synthesized images. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIV*, pages 146–163. Springer, 2022.

[6] Beijing Chen, Xingwang Ju, Bin Xiao, Weiping Ding, Yuhui Zheng, and Victor Hugo C de Albuquerque. Locally gan-generated face detection based on an improved xception. *Information Sciences*, 572:16–28, 2021.

[7] Beijing Chen, Xin Liu, Yuhui Zheng, Guoying Zhao, and Yun-Qing Shi. A robust gan-generated face detection method based on dual-color spaces and an improved xception. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(6):3527–3538, 2021.

[8] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[9] Davide Cozzolino and Luisa Verdoliva. Noiseprint: A cnn-based camera model fingerprint. *IEEE Transactions on Information Forensics and Security*, 15:144–159, 2019.

[10] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[11] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. In *International conference on machine learning*, pages 3247–3258. PMLR, 2020.

[12] Jessica Fridrich and Jan Kodovsky. Rich models for steganalysis of digital images. *IEEE Transactions on information Forensics and Security*, 7(3):868–882, 2012.

[13] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017.

[14] Sharath Girish, Saksham Suri, Sai Saketh Rambhatla, and Abhinav Shrivastava. Towards discovery and attribution of open-world gan generated images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14094–14103, 2021.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[16] Diego Gragnaniello, Davide Cozzolino, Francesco Marra, Giovanni Poggi, and Luisa Verdoliva. Are gan generated images easy to detect? a critical analysis of the state-of-the-art. In *2021 IEEE international conference on multimedia and expo (ICME)*, pages 1–6. IEEE, 2021.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[19] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23(47):1–33, 2022.

[20] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 101–117, 2018.

[21] Pedro Ribeiro Mendes Júnior, Luca Bondi, Paolo Bestagini, Stefano Tubaro, and Anderson Rocha. An in-depth study on open-set camera model identification. *IEEE Access*, 7:180713–180726, 2019.

[22] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

[23] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[24] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020.

[25] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021.

[26] Nitin Khanna, Aravind K. Mikkilineni, and Edward J. Delp. Scanner identification using feature-based processing and analysis. volume 4, pages 123–139, 2009. doi: 10.1109/TIFS.2008.2009604.

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[28] Matthias Kirchner. Fast and reliable resampling detection by spectral analysis of fixed linear predictor residue. In *Proceedings of the 10th ACM workshop on Multimedia and security*, pages 11–20, 2008.

[29] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022.

[30] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7824–7833, June 2022.

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. 2019.

[32] Owen Mayer and Matthew C Stamm. Forensic similarity for digital images. *IEEE Transactions on Information Forensics and Security*, 15:1331–1346, 2019.

[33] Owen Mayer and Matthew C Stamm. Exposing fake images with forensic similarity graphs. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):1049–1064, 2020.

[34] Owen Mayer, Belhassen Bayar, and Matthew C Stamm. Learning unified deep-features for multiple forensic tasks. In *Proceedings of the 6th ACM workshop on information hiding and multimedia security*, pages 79–84, 2018.

[35] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 405–421. Springer, 2020.

[36] Tomas Pevny and Jessica Fridrich. Detection of double-compression in jpeg images for applications in steganography. *IEEE Transactions on Information Forensics and Security*, 3(2):247–258, 2008. doi: 10.1109/TIFS.2008.922456.

[37] Alin C Popescu and Hany Farid. Exposing digital forgeries by detecting traces of resampling. *IEEE Transactions on signal processing*, 53(2):758–767, 2005.

[38] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.

[39] Andreas Rossler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Niessner. Faceforensics++: Learning to detect manipulated facial images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[40] Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[41] Matthew Stamm and KJ Ray Liu. Blind forensics of contrast enhancement in digital images. In *2008 15th IEEE International Conference on Image Processing*, pages 3112–3115. IEEE, 2008.

[42] Matthew C Stamm and KJ Ray Liu. Forensic detection of image manipulation using statistical intrinsic fingerprints. *IEEE Transactions on Information Forensics and Security*, 5(3):492–506, 2010.

[43] Eu Wern Teh, Terrance DeVries, and Graham W Taylor. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16*, pages 448–464. Springer, 2020.

[44] Amel Tuama, Frédéric Comby, and Marc Chaumont. Camera model identification with the use of deep convolutional neural networks. In *2016 IEEE International workshop on information forensics and security (WIFS)*, pages 1–6. IEEE, 2016.

[45] Luisa Verdoliva. Media forensics and deepfakes: an overview. *IEEE Journal of Selected Topics in Signal Processing*, 14(5):910–932, 2020.

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

[47] Guanshuo Xu, Han-Zhou Wu, and Yun-Qing Shi. Structural design of convolutional neural networks for steganalysis. *IEEE Signal Processing Letters*, 23(5):708–712, 2016.

[48] Yifeng Zhan, Yifang Chen, Qiong Zhang, and Xiangui Kang. Image forensics based on transfer learning and convolutional neural network. In *Proceedings of the 5th ACM workshop on information hiding and multimedia security*, pages 165–170, 2017.

[49] Baiwu Zhang, Jin Peng Zhou, Ilia Shumailov, and Nicolas Papernot. On attribution of deepfakes. *arXiv preprint arXiv:2008.09194*, 2020.

[50] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7):3142–3155, 2017.

[51] 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. doi: 10.1109/WIFS47025.2019.9035107.