Image-to-Image Translation Generative Adversarial Networks for Video Source Camera Falsification

Maryna Veksler[1(✉)], Clara Caspard[2], and Kemal Akkaya[1]

[1] Florida International University, Miami, FL 33174, USA
{mveks001,kakkaya}@fiu.edu
[2] Pomona College, Claremont, CA 91711, USA
cfcb2020@mymail.pomona.edu

Abstract. The emerging usage of multimedia devices led to a burst in criminal cases where digital forensics investigations are needed. This necessitate development of accurate digital forensic techniques which require not only the confirmation of the data integrity but also the verification of its origin source. To this end, machine and/or deep learning techniques are widely being employed within forensics tools. Nevertheless, while these techniques became an efficient tool for the forensic investigators, they also provided the attackers with novel methods for the data and source falsification. In this paper, we propose a simple and effective anti-forensics attack that uses generative adversarial networks (GANs) to compromise the video's camera source traces. In our approach, we adopt the popular image-to-image translation GANs to fool the existing algorithms for video source camera identification. Our experimental results demonstrate that the proposed attack can be implemented to successfully compromise the existing forensic methods with 100% probability for non-flat videos while producing the high quality content. The results indicate the need for attack-prone video source camera identification forensics approaches.

Keywords: Generative Adversarial Networks (GANs) · Multimedia forensics · Video Source Identification · Machine learning

1 Introduction

The rapid increase of Internet of Things (IoT) technologies triggered a massive impact on the digital forensics field due to the heavy involvement of such devices in crime science applications. Specifically, the popularity of multimedia content generated by various IoT devices such as phones, body cameras, drones, vehicles, etc. for information sharing and storage caused the forensics investigators to develop techniques that will enable thorough data validation and analysis [10]. At the same time, the improvements in machine learning (ML) and artificial

S. Goel et al. (Eds.): ICDF2C 2022, LNICST 508, pp. 3–18, 2023.
https://doi.org/10.1007/978-3-031-36574-4_1

intelligence (AI) algorithms provided criminals with sophisticated tools for media content alteration and forgeries [12,22,30].

One example use case where forgeries can be employed is multimedia forensics where one of the major problems is the identification of the source camera to validate the data origin. Given that identifying a video emerging from a specific device belonging to a victim or suspect might be crucial for the digital forensics investigation, it is vital to identify the source camera with high precision. Researchers developed numerous techniques aimed to determine the model of the camera devices used for image and video recording, based on the noiseprints introduced by the source camera. Proven to be unique, the camera noiseprints are the result of insignificant manufacturing defects present on camera lenses and can be extracted using statistical analysis [6,15] and deep learning (DL) [32] approaches.

These techniques, however, can be compromised by the attacker deploying various anti-forensics techniques [13,21,24,29] to disrupt the camera identification process. Thus, researchers and investigators require a comprehensive knowledge of anti-forensics to understand the weaknesses of the existing methods and incorporate the appropriate protection mechanisms [5,27,38].

One approach to fool video source camera identification techniques is to deploy a DL framework called *generative adversarial network* (GAN) to generate data that mimics camera-specific noise. GANs have become popular within the last several years with many applications [1–3,36]. In the context of video source camera identification, criminals might aim to generate fake videos as if these are coming from the phone camera of a specific victim. However, the existing GANs for source camera falsification mostly target images omitting the video content, as the latter requires complex processing of various elements (e.g., sequence of the video frames, bit-rate, and audio stream). Moreover, despite the wide variety of the video GANs [4] (i.e., generating fake videos based on the given input conditions), there is no demonstration of the GAN used to falsify video source camera origin.

In this paper, we explore the effectiveness of using GANs for the source camera origin falsification and the resistance of the existing video source camera identification (VSI) techniques against fake data. Specifically, we select a pre-trained video source camera identification framework proposed in [32] as a forensic classifier. Our choice of the VSI network is justified by its high accuracy compared to the other existing methods. Due to the complexity of the video streams, and the fact that most of the VSI approaches are applied to the patches extracted from the video frames, we apply the GANs on the patches directly. Therefore, we demonstrate that the adversary may fool the VSI networks by manipulating the selected patch data and does not require generating realistic video footage. Furthermore, we adopt the popular open-source image-to-image translation GANs to implement video source falsification techniques. Thus, we indicate that the video source falsification attack can be implemented given limited resources. Specifically, we select CycleGAN to generate fake data for two distinct video source cameras and then use the trained networks to generate the falsified video data for both devices.

We conducted a series of experiments to (1) assess the ability of GANs to trick the video source identification framework and (2) identify the performance of the VSI network in adversarial settings. The results demonstrate that CycleGAN can successfully fool the video source camera classifier with 66% and 100% probability for video content of any and non-flat types respectively. Moreover, the generated data preserves the original content, indicating that the accuracy of the selected VSI network is not affected by the objects present in the video.

The rest of the paper is organized as follows. First, we present the review of related work in Sect. 2. In Sect. 3 we provide the background on implemented video source camera identification model and GAN architecture, followed by the description of our methodology in Sect. 4. In Sect. 5, we describe the experimental results and summarize the key outcomes. Finally, we present the conclusions and highlight some future work in Sect. 6.

2 Related Work

The applications of multimedia GANs widely vary from improving the quality of scientific datasets [3] and entertainment development [1] to protecting users' privacy [36], and can be used to compromise forensics analysis of the data, such as source camera identification [7] and facial recognition [33]. These networks can be used to compromise the forensic analysis of the data.

General Applications of Multimedia GANs. The existing video GANs can be grouped into three distinct categories based on their architecture. The first type of video GANs uses recurrent neural networks (RNN) architectures to analyze time-series data and produce the final video via temporal correlations [34]. Tulyakov et al. [31] designed a Motion Content GAN that derives the fake video by mapping random vector sequences to the video frames sequences, where each random vector consists of content and motion parts. The second category of video GANs is referred to as progressive video GANs which generate frames first, and then another generator is used to convert frames into the video [14,23]. Finally, video GANs with two-stream architecture analyze different aspects of video and its frames. The video GAN developed in [35], contains two independent generator streams for background and foreground content. The results of these generators are then combined via a motion pathway mask to produce a short video.

Forensics Applications for Multimedia GANs. In the multimedia forensics field, the GANs can be used to disrupt the data analysis or to impose the modified data as authentic. In [25], the authors developed a BDC-GAN for bi-directional conversion between computer-generated and natural images, which allows bypassing the majority of existing forensic detectors. Zou et al. [38] applied GANs, which improves the image quality while staying undetected by the methods of contrast enhancement detection.

The recent work attempt to use GANs as a means to disrupt the forensics approaches for multimedia source camera detection. However, it primarily targets the image data omitting complex video content. Chen et al. [8] designed

an image source falsification GAN based on the Convolution Neural Network (CNN) architecture. In [9], the authors introduce a new component into the traditional GAN architecture, called the embedding network. This network is used to extract the camera noiseprint feature vector, that provides feedback to the generator based on the loss between falsified and real source camera images.

Difference from Existing Work. Unlike previous works, we focus on video forgeries that target video source camera identification frameworks. Therefore, our main aim is to replicate the insignificant noise traces unique to the camera device rather than the video contents. We develop a simple GAN attack on the VSI forensics techniques that can be implemented at a low resource and effort cost. We further transferred the video source camera falsification problem to the image level by applying the GAN directly to the patches extracted from the video frames. To the best of our knowledge, it is the first work to implement GAN-based anti-forensics video source camera falsification.

3 Background

In this section, we provide details about video source camera identification and the structure of GANs.

3.1 Video Source Camera Identification Network

Video source camera identification helps us identify the genuine source of an existing video among various cameras as shown in Fig. 1. The majority of the recent approaches use DL-based Convolutional Neural Networks (CNN) to extract the device-specific noiseprint across the set of videos [11,18,26,32]. Before being analyzed by CNN networks, the video data is processed such that the video intra-coded or I-frames are extracted and often split into patches of smaller size. For our implementation, we selected a recent video source camera identification approach developed in [32] as a benchmark since it is proven to achieve higher classification accuracy compared to other systems. The proposed network consists of four independent CNNs, where each extracts noiseprint from one of four non-overlapping quadrants of the video I-frames.

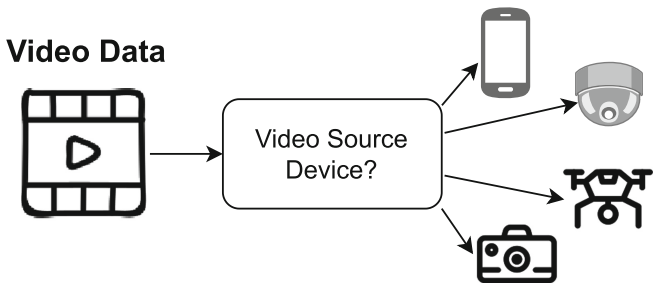

Fig. 1. Diagram illustrating video source camera identification process.

The video source camera is identified as follows. First, 15 I-frames are extracted from the video under the investigation, followed by dividing each frame into four non-overlapping patches of 128×128 size. Next, homogeneous patches are selected from each quadrant based on the standard deviation of the quadrant data. This techniques allows to determine the patches that may contain camera specific noise, while excluding the patches with needless amount of foreground content. Then, 4 pre-trained CNNs are used to identify the patches from different quadrant of the same video frame. The final stage of the ensemble CNNs utilizes average voting to aggregate the base learner CNN predictions for quadrant patches, followed by the majority voting to obtain video source camera prediction. The average voting produce the ensemble output vector representing the camera source prediction for the frame. After collecting the ensemble prediction for all video frames, the majority voting technique is applied to determine the video source camera label.

3.2 Generative Adversarial Networks (GANs)

First proposed in 2014 [17], generative adversarial networks (GANs) established a new category of DL algorithms. The capability to generate a realistic and highly accurate data from various domains such as text, images, video, and statistical measurements, triggered the wide adaptation of GANs in the industry and academia. Compared to other generative algorithms, GANs provides an efficient way to train the generator using a game theory principles, two-player zero-sum minimax game. Specifically, the GAN consists of two sub-models, referred to as a generator model trained to generate a new data and a discriminator model used to classify the data as either real or fake. These sub-models are set against each other with a simple goal to outperform the opponent. At the same time, the gain and loss are exactly balanced between the generator and discriminator, attempting to reach the total of zero utilization. The analogy of the process can be drawn by comparing the generator model to a counterfeiter who tries to create fake banking checks while the discriminator model is the authorities trying to accurately pick out those counterfeit checks.

Based on the unsupervised learning algorithm, GANs do not require a labeled or pre-processed dataset to generalize the data. Rather, the network attempts to devise the new samples by interpreting the feedback from the discriminator. The GANs are trained by taking any random type of input and generating a new data similar to the real samples provided to the discriminator.

The architecture of GAN sub-models is highly flexible and depends solely on the input data type. Moreover, the discriminator can be represented by any type of the classifier suited for the data type produced by the generator. Therefore, multimedia GANs usually implement CNN networks as a generative sub-model. On the other hand, the generator architecture is generally either that of the neural network (NN) or the CNN depending on the complexity of the task. However, most of the multimedia GANs employ the CNN architecture to either generate synthetic samples from a random noise or translate data from one domain to another. Figure 2 illustrates the generic architecture of an image GAN.

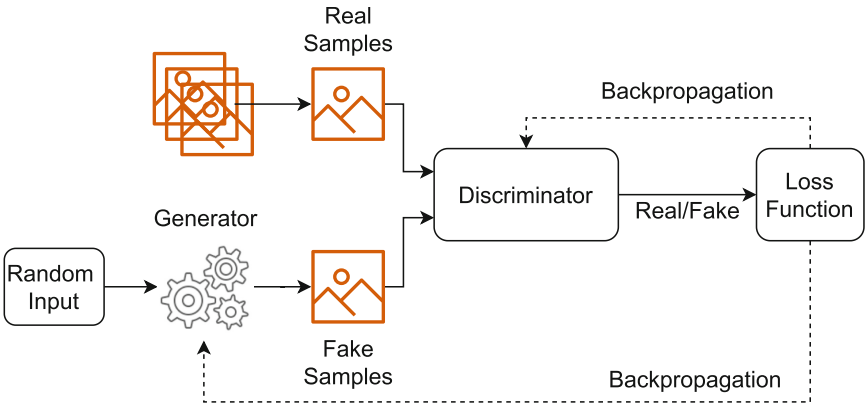

Fig. 2. General setup for GANs architecture and training.

4 Methodology

The goal of our work is to explore the viability of an anti-forensic attack against the video source camera identification method. Therefore, we define three primary objectives as follows:

- The GAN content should disrupt the accuracy of the video source camera identification framework;
- The GAN generator should mimic a specific video source camera noiseprint;
- The GAN data should look realistic and contain no visible artifacts of modifications.

To this end, we propose implementing the image-to-image translation GAN for video source camera falsification via applying the network to the patches extracted from the video frames as detailed below.

4.1 Selecting GAN Network

When selecting the optimal GAN architecture, we focused on the networks that can be easily implemented and do not require extensive knowledge of the video source camera identification framework. Moreover, since most of the existing video source camera identification mechanisms, including the one described in Sect. 3, often operate on video frames' patches of the sizes between 128×128 and 512×512, the GAN should be oriented toward generating smaller-sized images.

We examined various types of GAN architectures such as conditional GAN (CGAN) [16], progressive GAN [20], and image-to-image translation [19]. First, we rejected the CGAN architectures, as our objectives were not concerned with labels-based targeted image generation. Next, we eliminated progressive GANs since their primary goal is the generation of high-resolution images and the progressive GAN requires a large amount of computational power. Therefore, it

contradicts two of our goals; minimizing the complexity of the architecture and producing fake images with lower resolution.

As a result, we decided to pick the image-to-image translation (I2IT) GAN approach as the backbone of our method. Specifically, we selected the CycleGAN [37] architecture since, unlike traditional I2IT GANs, it does not require the predetermined pairing of the images from the opposite domains. Rather, the image pairs are detected automatically, which significantly reduces the pre-processing time. Thus, we can train the network on the limited data set with the indirectly correlating samples. Moreover, as CycleGAN is designed to operate on images with size 128×128 pixels in either three-channel (RGB) or gray-scale domains, its architecture does not require significant modifications when used for video source camera falsification.

4.2 GAN Architecture

CycleGAN consists of two generators and two discriminators sub-models. It accepts as an input a pair of images from two different domains, A and B, and transforms them into the data of the opposite domain. Figure 3 illustrates the architectural overview of both networks. Specifically, *generator AB* accepts the samples from Domain A and translates them into the Domain B. Next, *discriminator B* is used to classify the real and newly generated fake samples of the Domain B. At the same time, *generator BA* converts samples from Domain B to Domain A, with subsequent *discriminator A* as a classifier.

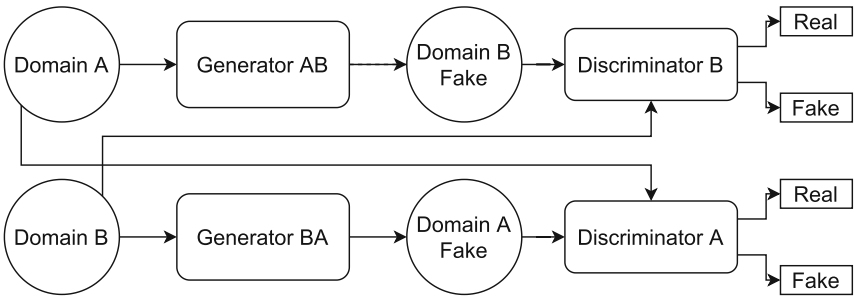

Fig. 3. Overview of the CycleGAN Architecture.

Generator. The inner architecture of the CycleGAN generator is given in Fig. 4. It consists of two phases down-sampling and up-sampling. First, the original image is downsampled by applying four convolutional layers with a stride 2 (i.e., the filters are advancing by the step of two pixels). After each convolutional layers, we apply LeakyReLU activation (A) followed by the Instance Normalization (N). We select Instance Normalization instead of Batch Normalization, as it allows us to normalize each sample independently instead of generalizing across multiple samples.

During the up-sampling process, the output of the previous layer passes through the *Up Sampling Block*, which consists of the simple UpSampling2D layer followed by the convolutional and instance normalization layers. Then the output of the *Up Sampling Block* is concatenated with the output of the corresponding downsampling convolutional layer. As a result, the generators produce an image of the same size as the Input layer.

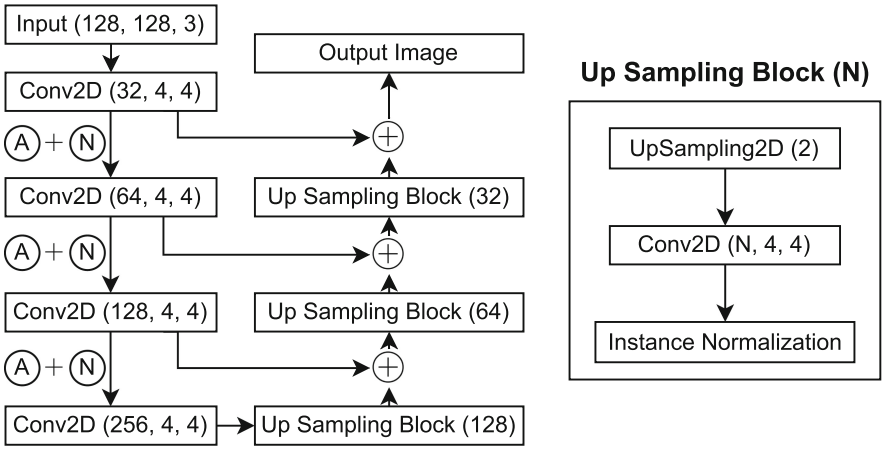

Fig. 4. CycleGAN Generator Architecture.

Discriminator. We select a simple discriminator architecture that consists of an input layer of the size $3 \times 128 \times 128$, four consecutive residual blocks, and an output layer. Each of the residual blocks contains a 2D Convolution layer, followed by LeakyReLU activation, and instance normalization. The first convolutional layer applies 64 filters of kernel size 4×4, while each consecutive convolution doubles the filter size. The design of the discriminator is driven by (1) a low number of convolutional layers to increase processing time and (2) high accuracy for images classification.

Loss Function. We used two types of loss functions, adversarial and cycle consistency. The adversarial loss is calculated using the mean square error (MSE) function and reflects the generators' attempts to successfully "fool" the discriminators. On the other hand, the cycle consistency, calculated as a mean absolute error (MAE) value, not only reflects the likelyhood of the generated data to be of the given domain but also asses the probability of the generators' input and output images to look the same. The CycleGAN uses a single cycle consistency loss to assess the performance of both generators. During the training process, the main objective of the GAN generators is to minimize the cycle-consistency loss.

5 Evaluation Results

We implemented the CycleGAN using Python scripting language to evaluate its effectiveness for video source camera falsification. Specifically, we selected the framework proposed in [32] as a benchmark for video source camera identification (VSI) and tested our hypothesis for the video content of two distinct phone models. During the experiments, we assessed the efficacy of the proposed approach based on:

- the requirements and limitations of the approach;
- the presence of the artifacts in the GAN generated data;
- the ability of the GAN-generated content to disrupt the functionality of the VSI framework and mimic specific video camera noise prints;
- the performance of the VSI network.

5.1 Dataset

We used the open-source VISION dataset [28] designed for multimedia forensics investigations to obtain the videos for our work. Specifically, we selected three phones of distinct models - Huawei P9, Apple iPhone 6, and Samsung Galaxy S5. For each device, we picked three types of videos containing flat, indoor, and outdoor scene content.

For each content type, we selected natively recorded videos and their corresponding WhatsApp and YouTube processed versions. This totaled to nine distinct video types per device. Therefore, for each device we selected 19 native videos, such that each video has its corresponding WhatsApp and YouTube version, resulting in a total of 57 videos per phone model. This collection of data provided us with adequately varied media content to train and test both GAN and VSI models. To train our GAN and VSI models, we split the dataset into training and test sets with a ratio of 70:30. As a result, we ended up with 18 test videos per device. To further prepare the dataset, we split each video into I-frames, subsequently divided into four non-overlapping quadrants. From each quadrant, we extracted 128×128 pixels patches, so that each video in our dataset was reduced to a series of a few thousand patches of images belonging to a certain area of the full frame at one specific moment of the video. Figure 5 demonstrates a complete process of video decomposition into the patches.

5.2 Experiments

Requirements and Limitations: We simplify the complex video generation problem by applying the transformation for video frame patches directly to implement the CycleGAN for video source camera falsification. To successfully train CycleGAN for falsification of the particular video camera source, the adversary should have access to the data from both the original and target devices. Nonetheless, if the main goal is to simply hide the fingerprints of the video camera source origin, the adversary only needs access to the original video and the

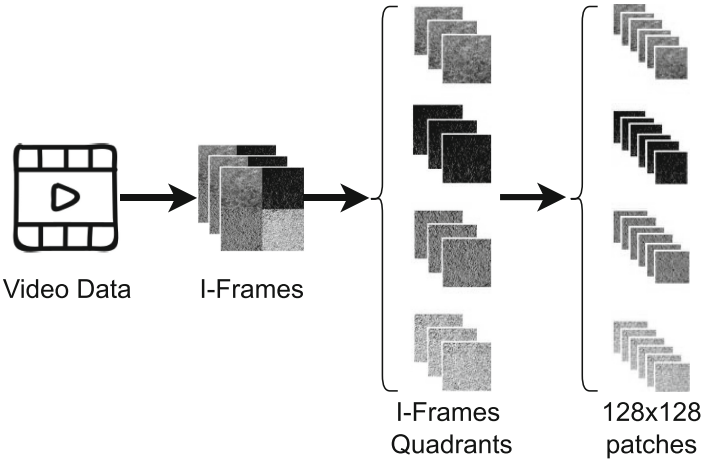

Fig. 5. Decomposition of the videos into 128×128 patches.

data from a set of random devices. In this case, the CycleGAN generators will change the trace of the source camera by overlapping them with information from other devices.

System Setup: The setup of our experiment consisted of two phases. First, we reproduced the ensemble CNNs framework [32] as a baseline for VSI process. We trained the network to classify three distinct devices and achieved the accuracy of 95.6% for patch level classification, resulting in all test videos to be classified correctly.

Next, we trained our GANs. We designed four independent CycleGANs corresponding to the data obtained from the four non-overlapping video frame quadrants. Unlike for VSI framework, we trained GANs using only Apple iPhone 6 and Samsung Galaxy S5 data, which corresponded to Domain A and Domain B. As a result, each GAN produced 2 distinct generators for Apple-to-Samsung and Samsung-to-Apple patch conversion. Figure 6 illustrates the flow of our experimental setup to impose the video data originating from an Apple device as belonging to the Samsung.

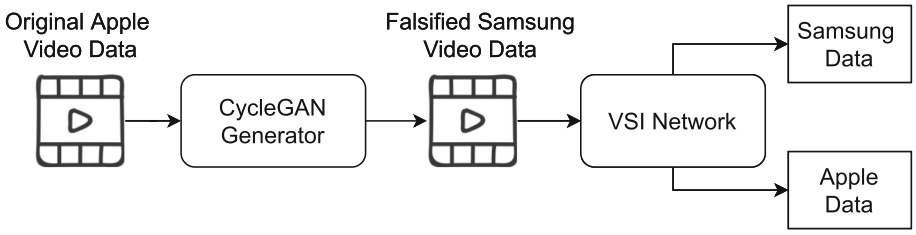

Fig. 6. The flow of the experimental setup for the video source camera falsification.

We used an Adam optimizer for the CycleGAN discriminators and generators with a learning rate set to 2e−4 and the exponential decay rate of 0.5. We conducted multiple rounds of experiments to determine the optimal settings for both networks that allow to achieve high network stability while minimizing the network training time. According to our tests, the network reaches the optimal performance with a batch size of 1 used to train the model for 15 epochs.

While we used a test dataset to evaluate the accuracy of the video source camera identification framework, the same test data was used as an input to the trained CycleGAN generators. This approach ensured that the generated video data have remains unknown to the source camera identification framework during the training stage. From each GAN architecture, we obtained a generator trained to translate Apple data into Samsung data, and a generator with the opposite function. Therefore, we trained eight CycleGAN generators in total.

Experimental Results: We conducted a total of two experiments. First, we used Samsung-to-Apple (S2A) generators to test the VSI network's ability to correctly classify CycleGAN-modified Samsung video data. Table 1 indicates the percentage of the videos classified as originating from one of the three devices. The results indicate that the VSI network successfully identified all of the fake video data as belonging to the correct origin device - Samsung Galaxy S5. Therefore, the S2A CycleGAN generator failed to fool the VSI network.

Table 1. The results of the VSI network for the data generated using the S2A CycleGAN generator (i.e., actual video data source is Samsung, fake video is generated as if this is from Apple), indicate the percentage of the videos identified originating from a given device.

	Huawei	Apple	Samsung
Flat videos	0.0%	**0.0%**	*100.00%*
Indoor videos	0.0%	**0.0%**	*100.00%*
Outdoor videos	0.0%	**0.0%**	*100.00%*

For the second experiment, we generated a new batch of video data using the Apple-to-Samsung (A2S) generator. The results of the VSI classification for the CycleGAN data are represented in Table 2. Unlike in the first experiment, the A2S generator achieved high success in imposing indoor and outdoor types of Apple video content as originating from the Samsung device. At the same time, VSI system was able to correctly identify the origin device for faked data containing flat scenes.

Artifacts Indicating the GAN Processing: One of the most important features of the GAN network when applied as an anti-forensics technique is its ability to hide the presence of artificial modification. Figure 7 illustrates video patches generated by CycleGAN generators. When analyzing the visible modification traces present, the video patches produced by both SA and AS generators

Table 2. The results of the VSI network for the data generated using the A2S CycleGAN generator (i.e., actual video data source is Apple, fake video is generated as if this is from Samsung), indicate the percentage of the videos identified originating from a given device

	Huawei	Apple	Samsung
Flat videos	0.0%	*100.00%*	**0.0%**
Indoor videos	0.0%	*0.0%*	**100.00%**
Outdoor videos	0.0%	*0.0%*	**100.00%**

have a pixelization noise. Nonetheless, the CycleGAN was able to mostly preserve the content of the original data. The main visual artifact which give away the manipulation appearance of intensified red and green pixelized lines as a content overlay.

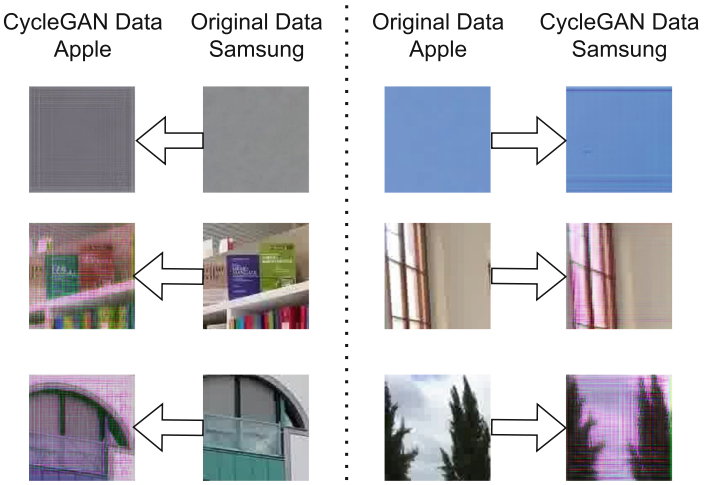

Fig. 7. The example of patches generated by CycleGAN for three types of video data based on the content - flat (top), indoor (middle), and outdoor(bottom). Left side: video patches produced by SA generator. Right side: video patches produced by AS generator.

CycleGAN Performance: Our experiments demonstrated that CycleGAN architectures can be successfully applied as an anti-forensics approach for video source camera identification. However, the results vary depending on the type of the original device and its content. While trained together using the same amount of training data and same setup, two generators exhibit very different results. Specifically, the SA generators failed to fool or partially disrupt the VSI network classification accuracy. On the contrary, the AS generator was able to

successfully falsify the video source camera for the videos containing indoor and outdoor content. Analyzing the realisticity of the data, CycleGAN generators successfully produced patches closely reflecting the original content. However, the resulting fake data contained noise artifacts which present a strong evidence to claim data manipulation. While these traces can be recognized during the manual analysis, it does not seem to influence the performance of VSI network specifically. Moreover, as a patch represents a smaller part of the video frame, it will not be recognized by a human eye easily.

VSI Network Performance: According to our analysis, the VSI network accuracy was partially affected by the proposed video source camera falsification attack. The network appears to recognize the original traces of the Samsung device better than the Apple device. Since for both cases, the content of fake data was close to the original with the visible noise artifacts present, we conclude that the network (1) does not consider the video content during the identification process and focuses on the source device fingerprints and (2) artificial noises do not have a direct impact on the system classification ability. Consequently, the outcome of this research indicates that a detailed analysis of the factors causing a given VSI network to recognize the traces of one source device better than the other is needed.

6 Conclusion and Future Work

In this work, we implemented the CycleGAN as the anti-forensics video source falsification technique and evaluated the resistance of the VSI ensemble CNNs framework against the adversarial attacks. Our results indicate that CycleGAN can be successfully applied to falsify the traces of video source device with the probability of 100% for the videos containing indoor or outdoor content. We also identify that CycleGAN generators do not have the same probability for the successful video source camera falsification, caused either by (1) flaws in the GAN or (2) VSI network inability to recognize the fingerprints of the video source camera with the equal precision. At the same time, our experiments indicate that the tested ensemble CNNs do not consider the visual video content when determining the video camera origin, while also is not disturbed by artificial noises.

For future work, we will focus on improving the performance of the GAN generators, via modifying the GAN architecture such that the content does not contain visible artifacts. Furthermore, we will explore the approaches to produce the complete video sequence with modified video source camera noiseprints. Finally, we will conduct an extensive analysis to identify the causes of the unequal performance of VSI and GAN networks for different device models.

Acknowledgements. Research was sponsored by the Army Research Office and was accomplished under Grant Number W911NF-21-1-0264. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research

References

1. Auto-painter: cartoon image generation from sketch by using conditional Wasserstein generative adversarial networks. Neurocomputing **311**, 78–87 (2018). https://doi.org/10.1016/j.neucom.2018.05.045
2. GANs for medical image analysis. Artif. Intell. Med. **109**, 101938 (2020). https://doi.org/10.1016/j.artmed.2020.101938
3. Super-resolution using GANs for medical imaging. Proc. Comput. Sci. **173**, 28–35 (2020). https://doi.org/10.1016/j.procs.2020.06.005. International Conference on Smart Sustainable Intelligent Computing and Applications Under ICITETM 2020
4. Aldausari, N., Sowmya, A., Marcus, N., Mohammadi, G.: Video generative adversarial networks: a review. ACM Comput. Surv. **55**(2), 1–25 (2022)
5. Barni, M., Chen, Z., Tondi, B.: Adversary-aware, data-driven detection of double JPEG compression: how to make counter-forensics harder. In: 2016 IEEE International Workshop on Information Forensics and Security (WIFS), pp. 1–6 (2016). https://doi.org/10.1109/WIFS.2016.7823902
6. Chen, C., Stamm, M.: Robust camera model identification using demosaicing residual features. Multimed. Tools Appl. **80**, 1–29 (2021). https://doi.org/10.1007/s11042-020-09011-4
7. Chen, C., Zhao, X., Stamm, M.C.: MISLGAN: an anti-forensic camera model falsification framework using a generative adversarial network. In: 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 535–539 (2018). https://doi.org/10.1109/ICIP.2018.8451503
8. Chen, C., Zhao, X., Stamm, M.C.: Generative adversarial attacks against deep-learning-based camera model identification. IEEE Trans. Inf. Forensics Secur. **PP**, 1 (2019). https://doi.org/10.1109/TIFS.2019.2945198
9. Cozzolino, D., Thies, J., Rössler, A., Nießner, M., Verdoliva, L.: SpoC: spoofing camera fingerprints (2019)
10. Cozzolino, D., Verdoliva, L.: Multimedia forensics before the deep learning era. In: Rathgeb, C., Tolosana, R., Vera-Rodriguez, R., Busch, C. (eds.) Handbook of Digital Face Manipulation and Detection. ACVPR, pp. 45–67. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-87664-7_3
11. Dal Cortivo, D., Mandelli, S., Bestagini, P., Tubaro, S.: CNN-based multi-modal camera model identification on video sequences. J. Imag. **7**(8), 135 (2021)
12. Damiani, J.: A voice deepfake was used to scam a CEO out of $243,000 (2019). https://www.forbes.com/sites/jessedamiani/2019/09/03/a-voice-deepfake-was-used-to-scam-a-ceo-out-of-243000/?sh=34e8298a2241
13. Das, T.K.: Anti-forensics of JPEG compression detection schemes using approximation of DCT coefficients. Multimed. Tools Appl. **77**(24), 31835–31854 (2018)
14. Duan, B., Wang, W., Tang, H., Latapie, H., Yan, Y.: Cascade attention guided residue learning GAN for cross-modal translation (2019)
15. Flor, E., Aygun, R., Mercan, S., Akkaya, K.: PRNU-based source camera identification for multimedia forensics. In: 2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI), pp. 168–175 (2021). https://doi.org/10.1109/IRI51335.2021.00029

16. Gauthier, J.: Conditional generative adversarial nets for convolutional face generation (2015)
17. Goodfellow, I., et al.: Generative adversarial nets. Advances in Neural Information Processing Systems, vol. 27 (2014)
18. Hosler, B., et al.: A video camera model identification system using deep learning and fusion. In: ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 8271–8275 (2019). https://doi.org/10.1109/ICASSP.2019.8682608
19. Jeong, S., Lee, J., Sohn, K.: Multi-domain unsupervised image-to-image translation with appearance adaptive convolution (2022)
20. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability, and variation (2017)
21. Kirchner, M., Bohme, R.: Hiding traces of resampling in digital images. IEEE Trans. Inf. Forensics Secur. **3**(4), 582–592 (2008)
22. Korshunova, I., Shi, W., Dambre, J., Theis, L.: Fast face-swap using convolutional neural networks (2016)
23. Li, Y., Min, M.R., Shen, D., Carlson, D., Carin, L.: Video generation from text. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, AAAI 2018/IAAI 2018/EAAI 2018. AAAI Press (2018)
24. Mayer, O., Stamm, M.C.: Countering anti-forensics of lateral chromatic aberration. Association for Computing Machinery, New York, NY, USA (2017)
25. Peng, F., Yin, L., Long, M.: BDC-GAN: bidirectional conversion between computer-generated and natural facial images for anti-forensics. IEEE Trans. Circ. Syst. Video Technol. **32**, 1 (2022). https://doi.org/10.1109/TCSVT.2022.3177238
26. Rong, D., Wang, Y., Sun, Q.: Video source forensics for IoT devices based on convolutional neural networks. Open J. Internet Things (OJIOT) **7**(1), 23–31 (2021)
27. Sharma, S., Ravi, H., Subramanyam, A., Emmanuel, S.: Anti-forensics of median filtering and contrast enhancement. J. Vis. Commun. Image Represent. **66**(C), 102682 (2020)
28. Shullani, D., Fontani, M., Iuliani, M., Alshaya, O., Piva, A.: Vision: a video and image dataset for source identification. EURASIP J. Inf. Secur. **2017**, 15 (2017). https://doi.org/10.1186/s13635-017-0067-2
29. Stamm, M.C., Lin, W.S., Liu, K.J.R.: Temporal forensics and anti-forensics for motion compensated video. IEEE Trans. Inf. Forensics Secur. **7**(4), 1315–1329 (2012). https://doi.org/10.1109/TIFS.2012.2205568
30. Thies, J., Zollhöfer, M., Stamminger, M., Theobalt, C., Nießner, M.: Face2Face: real-time face capture and reenactment of RGB videos, vol. 62, no. 1 (2018)
31. Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: MoCoGAN: decomposing motion and content for video generation (2017)
32. Veksler, M., Aygun, R., Akkaya, K., Iyengar, S.: Video origin camera identification using ensemble CNNs of positional patches. In: 2022 IEEE 5th International Conference on Multimedia Information Processing and Retrieval (IEEE MIPR) (2022). (in Press)
33. Venkatesh, S., Zhang, H., Ramachandra, R., Raja, K., Damer, N., Busch, C.: Can GAN generated morphs threaten face recognition systems equally as landmark based morphs? - vulnerability and detection (2020)
34. Villegas, R., Yang, J., Hong, S., Lin, X., Lee, H.: Decomposing motion and content for natural video sequence prediction. ArXiv abs/1706.08033 (2017)

35. Vondrick, C., Pirsiavash, H., Torralba, A.: Generating videos with scene dynamics. In: NIPS 2016, pp. 613–621. Curran Associates Inc., Red Hook, NY, USA (2016)
36. Yu, J., Xue, H., Liu, B., Wang, Y., Zhu, S., Ding, M.: GAN-based differential private image privacy protection framework for the internet of multimedia things. Sensors **21**(1), 58 (2021)
37. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2242–2251 (2017). https://doi.org/10.1109/ICCV.2017.244
38. Zou, H., Yang, P., Ni, R., Zhao, Y., Zhou, N.: Anti-forensics of image contrast enhancement based on generative adversarial network (2021)