DETECTING GAN-GENERATED IMAGERY USING SATURATION CUES

Scott McCloskey and Michael Albright

Honeywell ACST
Golden Valley, MN, USA
{scott.mccloskey, michael.albright}@honeywell.com

ABSTRACT

Image forensics is an increasingly relevant problem, as it can potentially address online disinformation campaigns and mitigate problematic aspects of social media. Of particular interest, given its recent successes, is the detection of imagery produced by Generative Adversarial Networks (GANs), e.g. 'deepfakes'. Leveraging large training sets and extensive computing resources, recent GANs can be trained to generate synthetic imagery which is (in some ways) indistinguishable from real imagery. We analyze the structure of the generating network of a popular GAN implementation [1], and show that the network's treatment of exposure is markedly different from a real camera. We further show that this cue can be used to distinguish GAN-generated imagery from camera imagery, including effective discrimination between GAN imagery and real camera images used to train the GAN.

1. INTRODUCTION

With the increasing importance of social media as a means of disseminating news, online disinformation campaigns have gotten significant attention in recent years. The dated phrase 'seeing is believing' is still descriptive of how people validate such stories, though, so image forensics is increasingly important. While social media makes the dissemination of fake news easier, computer vision tools have contributed to this trend by making it easier to generate fake imagery. Whereas an image manipulator in prior years would need significant experience with rendering and/or image manipulation software, modern data-driven approaches have made it much easier to generate artificial imagery from scratch.

Our paper concerns the development of forensics to detect imagery from Generative Adversarial Networks (GANs). While these are indistinguishable from real imagery *to the GAN's discriminator*, they differ in important ways from images taken by a camera. We analyze the structure of the generator network, paying particular attention to how it forms *color*, and note important differences. Specifically, the generator's internal values are normalized to constrain the outputs, in a way which limits the frequency of saturated pixels. We investigate the effectiveness of this cue in detecting two types of GAN imagery: one being imagery wholly generated by a GAN and the other where GAN-generated faces replace real faces in a larger image. Based on this, we compare our performance to the GAN discriminator, demonstrating that we can successfully differentiate between the generator's output and real images on which the GAN was trained.

Fig. 1. Example artifacts evident in GAN-generated imagery. Top image shows checkerboard artifacts introduced by deconvolution steps. Bottom image shows mismatched eye colors, similar to a cue used in existing forensics [2].

Our approach is based on an analysis of the GAN generator architecture, and is complementary to approaches that detect visual artifacts in the synthesized imagery. While these two types of approaches are complementary, the rapid pace of GAN developments will likely mitigate the effectiveness of artifact-based detection. In particular, the checkerboard artifacts illustrated in Fig. 1 have already been mitigated [3] by replacing deconvolution steps with up-sampling followed by convolution steps. Additional cues, such as mismatched eye colors or a lack of blinking in GAN-generated video [2] are similarly likely to be eliminated.

This material is based upon work supported by the United States Air Force and the Defense Advanced Research Projects Agency under Contract No. FA8750-16-C-0190. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force or the Defense Advanced Research Projects Agency.

2. RELATED WORK

Since their introduction in 2014 [4], GANs have quickly become an extremely valuable tool in a range of computer vision applications. At a high level, the concept of a GAN is that two networks are trained to compete with one another. The 'generator' network is trained to produce artificial imagery that is indistinguishable from a given dataset of real imagery, whereas the 'discriminator' is trained to correctly classify imagery as being either real or coming from the generator. Early attempts at this [5] were able to generate convincing imagery of simple image datasets such as MNIST digits [6], but had a harder time mimicking more complicated images. More recently, computational techniques have been introduced which can generate convincing facial imagery [1] and have increased the resolution of generated imagery [7].

In response to the development of GANs, the forensics community has begun to develop methods to detect whether or not a given image was generated by a network trained in a GAN framework (for brevity, we refer to the detection targets as 'GAN images', and seek to distinguish them from 'real images'). One such method [2] uses the lack of blinking in DeepFake-type videos to detect GAN videos. Other approaches, rather than leveraging semantically-meaningful cues, use machine learning and neural networks to distinguish GAN from real images. Marra et al. [8] use a network based on XceptionNet, Hsu et al. [9] develop a deep forgery discriminator with a contrastive loss function, and Guera and Delp [10] use recurrent neural networks to detect GAN video. A key concern with methods based on deep networks is that they could easily be incorporated into the GAN's discriminator and, with additional training, the generator could be fine-tuned in order to learn a counter-measure for any differentiable forensic. An exclusively learning-based approach also makes it difficult to explain the outputs of the network, which is necessary in some forensics applications. Our approach, which is complementary to the above-mentioned forensics, is to analyze the structure of the GAN's generator and see how it impacts image statistics.

3. GAN GENERATOR ARCHITECTURE

In this section, we review the network architecture of the GAN's generator, and propose a specific cue which can be used to distinguish GAN imagery from real imagery. Since generators use different structures from one GAN to another, we look at features which are common among GANs. Also, in order to improve the detectability, we focus our attention on the later layers of the generator, since cues introduced in these layers are less likely to be modulated by subsequent processing.

Figure 2 shows a representative generator architecture, which expands a learned feature representation of the desired class to an image of such an object. The last layer of the generator, in particular, produces a 3-by-W-by-H output array: an image having 3 color channels, W columns, and H rows. The *input* to this last layer is an array of size K-by-W-by-H, where the $K > 3$ layers are referred to as 'depth' layers. The conversion of the K depth layers to the red, green, and blue color channels provide potential forensics cues, particularly in how they differ from camera-based image formation.

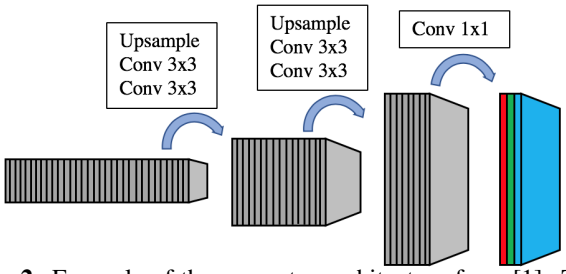

Fig. 2. Example of the generator architecture from [1]. The high-resolution image is produced from an input 'latent vector' by repeated upsampling (doubling the spatial dimensions), followed by 3x3 convolutions with leaky-ReLU activations and pixel-wise normalization. The final color image is generated by a 1x1 convolution.

A common process in GAN generators is the use of normalizations, which are employed in order to encourage convergence in training. As with color image formation, the exact method of normalization varies from one GAN to the next. In [1], for example, pixel-wise normalization is applied after convolution layers, so that the values of the 'depth' vector at each pixel have a fixed magnitude, i.e.

$$b_{j,x,y} = \frac{a_{j,x,y}}{\sqrt{\frac{1}{N}\sum_{c=0}^{N-1}(a_{c,x,y})^2 + \epsilon}}, \tag{1}$$

where a is an unnormalized feature map, b is the pixel-wise normalized version, indices x and y denote the spatial location of the pixel, indices j and c denote the depth position in the feature map, N denotes the number of feature maps, and $\epsilon = 10^{-8}$. In [7], normalization is applied within the individual 'depth' planes as,

$$b_{n,c,x,y} = \gamma_{n,c}\left(\frac{a_{n,c,x,y} - \mu_{n,c}}{\sigma_{n,c}}\right) + \beta_{n,c} \tag{2}$$

$$\mu_{n,c} = \frac{1}{HW}\sum_{x=0}^{W-1}\sum_{y=0}^{H-1} a_{n,c,x,y} \tag{3}$$

$$\sigma_{n,c} = \sqrt{1/HW\sum_{x=0}^{W-1}\sum_{y=0}^{H-1}(a_{n,c,x,y} - \mu_{n,c})^2 + \epsilon}, \tag{4}$$

where b and a are the normalized and unnormalized feature maps (respectively), x and y specify the spatial position of the pixel, c denotes the depth channel, n indexes the image in the mini-batch, and the parameters β and γ are learned during the

Fig. 3. Example images (top row) and grayscale histograms (bottom row) for two real images (left, right) and one GAN image (center) from [1]. Whereas the real images feature regions of under- or over-exposure (left and right images, respectively), GAN images (e.g., center) lack regions of saturation even when the background is white.

training process to constrain the mean and variance of values within the feature map depth planes.

Regardless of whether the normalization is applied pixel- or layer-wise, the result of both steps will be to have a relatively uniform distribution in the unit interval. These well-behaved values are then transformed into RGB intensities via a K-by-1 convolution. In camera-based imaging, however intensity values are not nicely constrained. Instead, irradiance values incident on a camera's sensor generally have a logarithmic distribution, necessitating high dynamic range (HDR) imaging [11]. HDR imaging involves the capture of multiple images separated by one or more *stops* (binary orders of magnitude) of exposure, e.g. images exposed for 1/15, 1/30, and 1/60 second. Without HDR, camera images generally have regions of saturation and/or under-exposure, as shown in Fig. 3. Because of the normalizations applied in the generator, however, GAN images lack these regions.

4. DETECTION METHOD

In this section, we propose a simple detection method based on the above analysis, in order to understand the predictive power of this cue. Given a relatively small set of training data, it is necessary to utilize pre-trained models (where applicable) or to use lower dimensionality features that can be trained with the data on hand.

For this forensic, the hypothesis is that the frequency of saturated and under-exposed pixels will be suppressed by the generator's normalization steps. This suggests a straightforward GAN image detector, where we simply measure the frequency of saturated and under-exposed pixels in each image. Specifically, for over-exposed pixels we measure a set of features

$$f_i^o = \frac{1}{HW} \|\{(x,y) \mid I(x,y) \geq \tau_i^o\}\|, \tag{5}$$

for $\tau_i^o \in \{240, 245, 250, 255\}$ (for an 8 bit representation of pixel intensities). Similarly, the frequency of under-exposed pixels are measured as features

$$f_i^u = \frac{1}{HW} \|\{(x,y) \mid I(x,y) \leq \tau_i^u\}\|, \tag{6}$$

for $\tau_i^u \in \{0, 5, 10, 15\}$.

These features are classified by a linear Support Vector Machine (SVM), trained using Matlab's fitcsvm function. The training data consisted of features from 1387 GAN-generated images (randomly sub-sampled from the 30 LSUN [12] categories of images created by the GAN in [1]), and real camera images from the ImageNet dataset.

5. EVALUATION

In order to evaluate the effectiveness of our detection method, we experimented with two benchmark datasets produced in conjunction with the US National Institute of Standards and Technology's Media Forensics Challenge 2018 [13]. The two datasets address two different sets of GAN imagery:

1. **GAN Crop** images represent smaller image regions which are either entirely GAN-generated or not.

2. **GAN Full** images are mostly camera images, but some faces have been replaced by a GAN-generated face, similar to deep fakes.

For both datasets, we compute the saturation count feature over the entire image, even though GAN Full images have small manipulated regions around faces. Following on the convention, we present our detectors' performance via a Receiver Operator Characteristic (ROC) curve, showing the true detection and false alarm rate as a function of a decision threshold applied to the continuously-varied score output by

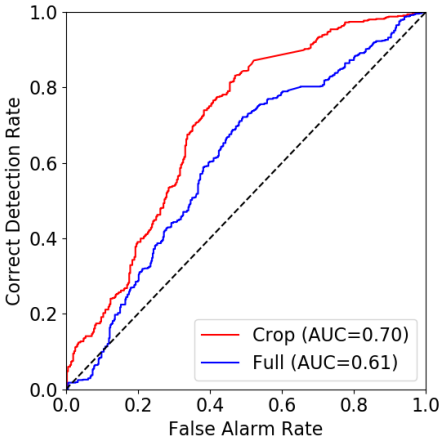

Fig. 4. ROC curves showing the performance of the saturation frequency SVM on the two GAN datasets from [13].

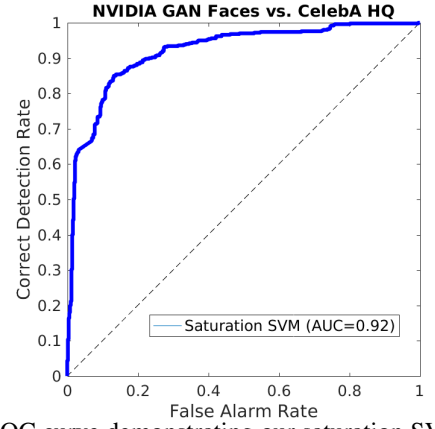

Fig. 5. ROC curve demonstrating our saturation SVM's ability to distinguish GAN-generated faces from the CelebA HQ face images used in [1]. Whereas the GAN's discriminator would have a 0.5 AUC and a diagonal ROC, our SVM can tell them apart.

each classifier. In an ROC curve, the performance of a random classifier would be a diagonal line. We also summarize the ROC with its Area Under the Curve (AUC), which would be 0.5 for a random detector and 1 for a perfect detector.

Figure 4 shows the ROC curve for our SVM trained on over-exposure features f^o. For both datasets, the performance is significantly better than a random detector. The method clearly does a better job of detecting fully GAN-generated images, where it produces a 0.7 AUC. In part this follows from a better match to the images used in training, but also the features' measuring a *proportion* of saturated pixels will be further diluted by the non-GAN regions in the GAN Full images. Despite this, the method still produces a respectable ROC and 0.61 AUC on the GAN Full image set.

5.1. Comparing to the GAN Discriminator

As a further experiment, we tested the performance of our saturation SVM in the same way that the GAN discriminator is evaluated. In [1], the GAN's discriminator is trained to differentiate real images of a certain category from the outputs of the generator trained to produces images in that category. As with most GANs, the generator wins the adversarial game, and ends up making images that are indistinguishable to the discriminator, i.e. the discriminator's ROC would be a diagonal line and the AUC would be 0.5. How much better does our SVM do in this setup? To test this, we look specifically at the case of face images, where the generator is trained with real images from a dataset called 'CelebA HQ', which is generated from the CelebA dataset [14] using rectification and super-resolution to 1k-by-1k pixels. The data from [1] include 1000 such images and 1000 images from their generator. Using the f^o features computed over both of these, we divide them into two equal sized training and testing sets. Fig. 5 shows the ROC curve on the test data, which demonstrates that our SVM does a very good job of distinguishing between the two set of images that are identical to the GAN's discriminator, having an AUC of 0.92.

6. CONCLUSION

We have described and evaluated the efficacy of a forensic related to the way in which generator networks from GANs transform feature representations to red, green, and blue pixel intensities. We demonstrated, in particular, that a relatively simple forensic based on the frequency of over-exposed pixels provides good discrimination between GAN-generated and camera imagery, via experiments with an independently-generated challenge dataset. Our method does quite well distinguishing fully GAN-generated image from natural images, and that it still provided some discrimination in the more difficult case where GAN-generated faces are spliced into a larger camera image. Lastly, we showed that our simple SVM detector does a very good job at what the GAN's discriminator couldn't do, namely tell CelebA HQ images from images produced by the GAN's generator.

That said, the pace of GAN-related innovation is quite high, and it's hard to predict how generators will be structured in the years ahead. We hope that the type of analysis here could encourage similar analyses of future GANs, in order to help address an important problem impacting increasingly large parts of society.

7. REFERENCES

[1] 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.

[2] Yuezun Li, Ming-Ching Chang, and Siwei Lyu, "In ictu oculi: Exposing ai created fake videos by detecting eye blinking," in *International Workshop on Information Forensics and Security*, 2018.

[3] Augustus Odena, Vincent Dumoulin, and Chris Olah, "Deconvolution and checkerboard artifacts," *Distill*, 2016.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., pp. 2672–2680. Curran Associates, Inc., 2014.

[5] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds., pp. 2234–2242. Curran Associates, Inc., 2016.

[6] Yann Lecun, Lon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradient-based learning applied to document recognition," in *Proceedings of the IEEE*, 1998, pp. 2278–2324.

[7] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional gans," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[8] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, "Detection of gan-generated fake images over social networks," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2018.

[9] C.-C. Hsu, C.-Y. Lee, and Y.-X. Zhuang, "Learning to Detect Fake Face Images in the Wild," *ArXiv e-prints*, Sept. 2018.

[10] D. Guera and E. J. Delp, "Deepfake video detection using recurrent neural networks," in *IEEE International Conference on Advanced Video and Signal-based Surveillance (to appear)*, 2018.

[11] Erik Reinhard, Greg Ward, Sumanta Pattanaik, and Paul Debevec, *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting (The Morgan Kaufmann Series in Computer Graphics)*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.

[12] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "LSUN: Construction of a Large-Scale Image Dataset using Deep Learning with Humans in the Loop," *ArXiv e-prints*, June 2016.

[13] National Institutes of Standards and Technology, "Media forensics challenge," https://www.nist.gov/itl/iad/mig/media-forensics-challenge-2018.

[14] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang, "Deep learning face attributes in the wild," in *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.