A Survey of DeepFake Detection Techniques

Tom Himler
Department of Computer Science
Virginia Tech
Blacksburg, VA 24060
thimler9@vt.edu

Abstract

With the rise of deepfake technology, threats to political and societal infrastructure are at stake, as the ease of creating effective misinformation increases. To ameliorate the potential impact, research and develop is being conducted to create models that can dissiminate between real and fake images and videos. This paper serves as a survey to these methods of deepfake detection. It finely looks at current research and dissiminates the benefits and deficits between methods for image and video forms of deepfakes detection. Finally, where current research is lacking and could be further studied will be mentioned.

1 Introduction

Deepfakes, a portmanteau of deep (for deep-learning) and fake, can be split into two classifications: *lip-sync* and *puppet-master*. Lip-sync deepfakes are videos of a target person mouthing the words of user-supplied audio. A famous example of this is the deepfake of Barack Obama being mouthed by Jordan Peele detailing the concerns of deepfakes [1]. Puppet-master deepfakes are videos that take a desired person and superimpose them onto another person to create a video that has the desired person acting as if they are another person. These are becoming common as a camera filter in Snapchat and the Chinese social media app, Zao, where users can insert themselves into famous TV and movie scenes [2].

The essence of deepfakes lies within a topic of Machine Learning called autoencoders and generative adversarial networks (GAN). Generally, deepfakes are formed by creating an autoencoder for the source image and an autoencoder for the target image. When an image gets processed through the encoder of the autoencoder, a compressed form of the source image called the latent representation is received. The latent representation of the image gets decoded through the target image's decoder, giving a naive form of a deepfake. Briefly, a GAN can be used as a sort of feedback loop that takes the deepfake images from the autoencoders and tries to determine if the image is faked or not in a neural network called a discriminator. The autoencoder is updated to improve until the discriminator can no longer tell the difference between real or fake for the input image or video.

With the rise of deepfakes comes an apprehension for the ethical implications. Deepfakes can easily be used as a tool to stir political or religious unrest, as it is easy to spread misinformation. Alongside that, deepfakes have been used a means of intimidation and blackmail since they can be used for deepfake pornography [3]. Deepfakes can be used to enable fraud by allowing individuals to assume identities of deceased people [4]. Surprisingly, a psychological impact of deepfakes is that people have even been shown to create false memories of events after watching deepfake videos [5], which further emphasizes the spread of misinformation. As deepfakes are so accessible and so capable of creating misinformation, there is widespread initiative of researching and developing tools to detect them.

Deepfake detection is a binary classification problem that takes a give image or video and determines if it is real or fake. Deepfake detection categories can be summed up into *image detection* and *video detection*. Image detection focuses on single frame deepfakes and takes advantage of their low-level pixel discrepancies or unique deepfake tool footprint, for example. Video detection looks at deepfake videos, using techniques that capture inaccuracies in the combination of spatial domain (single frame data) and temporal domain (relationship between frames data). This survey hopes to provide insight into current research on both categories of deepfake detection. Furthermore, the survey will provide insights on how different methods compare to one another, their benefits, their deficits, and what research still needs to be done.

2 Research Challenges

As deepfakes and deepfake detection methods form a growing battle between each other, research has been steered towards multiple directions to obtain the upper hand. Early on, methods for detecting deep fakes focused on honing in on artifacts that were created as part of the deepfake production process [6]. These days, researchers realized that in order to create more accurate models, deep learning approaches had to be studied. As a result, approaches that take advantage of convolutional neural networks (CNN) and long-term recurrent convolutional networks (LRCN) have been developed to identify discrepancies between the real and deepfake media. Many approaches take advantage of signals or data that deepfakes cannot properly replicate. Some researchers believe, however, that methods like these which focus more on the deepfake tool rather than the raw image leads to algorithms that become outdated quickly. As a result, new approaches try to find a sort of "fingerprint" that matches an image or video to the deepfake tool they were generated by. This leads to a deepfake tool-agnostic detection method that may be more resilient to unseen deepfakes and future deepfake generators.

The type and quantity of training data is also a key issue that plagues many of the deepfake detection papers. As with most deep learning models, the quality and relevance of training data is pertinent to worthwhile results. Because the field grows so quickly, it is hard to know if model performances are accurate. Some papers like [7], take a common data set (like CELEBA) and process it through the strongest available deepfake tool available. Others use reference datasets like FaceForensics++ [8]. Sometimes there isn't enough data available, so they have to make their own dataset, like what Agarwal et al. does in [9]. With no unified training data, it may be hard to directly compare the results between methods, so it's important to understand that these methods may vary in their true accuracies. Alongside that, I try to present a variety of current research to mitigate outdated data.

3 Deepfake Detection Methods

In this section methods of deepfake image and video detection will be presented. Generally, most detection methods use a neural network called a *convolutional neural network* (CNN) which is a neural network architecture that specializes in imagery. The goal of a CNN is to capture the essence of a picture without having to process every pixel through a neural network, as that is computationally expensive. Instead, it applies an operation called a filter that convolves through the image to find patterns that match the filter. This is repeated multiple times through a set of *convolutional layers* which get piped into a fully connected neural network for classification. For deepfake image detection, many methods take the idea of using CNN with a training set of deepfakes to find intrinsic characteristics that differentiate the deepfakes from real images or videos.

3.1 Deepfake Image Detection

In [10], Xuan et al. use pre-processing to remove high and low frequency impressions on images generated through a GAN. The goal of this method is to allow the CNN to learn more intrinsic information about the images. In general, the improvement of accuracy fluctuated depending on the dataset, but Xuan et al. achieved a maximum of 10% improvement. This form of preprocessing is

very common among other methods as a way of accounting for attacks that may take advantage of how the detector analyzes the images or videos. In this paper, they show that using Guassian blur and Guassian noise can improve accuracy, but as shown later, other methods change the contrast or color, apply rotations or mirroring, and apply compression methods to the source materials to improve training proficiency.

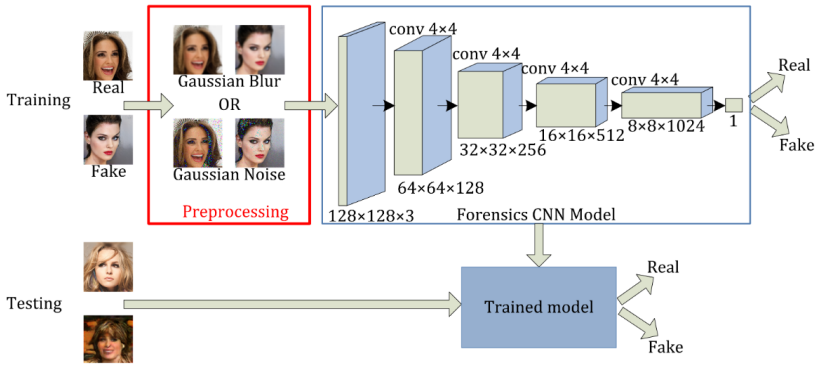

Figure 1: Pipeline for detection from [10]. Notice the preprocessing step in the training process. This method is prevalent among many modern techniques to improve accuracy.

Guo et Al. used a CNN model against deepfakes created by the Glow model [11]. They created a dataset from fakes created by the Glow model as training data for their CNN. Alongside that, they included training data that applied distortion operations to the images in hopes to increase the accuracy. Through this, they achieved a peak of 95.92% accuracy against the forgeries produced by the Glow model. Furthermore, the model also saw good accuracy against deepfakes from other deepfake models. As the Glow model is more state-of-the-art, it is effective at generalizing to other models, but there is a concern that this method will become outdated as deepfake generators become more advanced.

Guidice et al. present an alternative approach that relies on the Discrete Cosine Transform (DCT) to create a frequency spectrum fingerprint for analysis [7]. Instead of a black-box neural network solution, the algorithm uses a feature vector consisting of scale parameters of the Laplace Distribution for the AC components in the DCT of the image. They then compute a unique value called the GSF that represents the GAN (or no GAN for real deepfakes) that image is classified under. According to their analysis, different GANs tend to have differing GSFs compared to a real image. This approach is much more efficient than other common methods as its training cost is meager to deep neural networks. Alongside that, the process is very clearly understood, and as a result different attacks that attempt to circumvent the system can be analyzed. For example, Giudice et al. show that the algorithm is resilient against image transformations such as rotating, mirroring, and random square attacks. This is because the frequency fingerprint does not change or only slightly change after the transformation. In contrast, scaling, compressing, and Gaussian filtering do affect the accuracy. Without any form of attack, this model achieves a 99% accuracy against numerous datasets, and drops as low as 70% accuracy when faced against harsh JPEG compression.

In [8], Guarnera et al. describe a model that finds a convolutional trace (CT) associated with the input image. This convolutional trace is computed using an Expectation-Maximization algorithm to find correlations between neighboring pixels. They then used numerous models that take the CT of a given image to predict its efficacy. Among the models, the Radial Basis Function SVM and Random Forest models achieved the best accuracy of 91% and 98% respectively against a plethora of datasets. They also found that their methods also were robust to attacks such as random rectangles, Gaussian blur, rotations, scalings, and compressions. In fact, their method actually saw a boost in performance from a 90 degree rotation, surprisingly. They concluded that this may have been a result of the major direction the convolutions took place, and believe it should be studied in the future.

3.2 Deepfake Video Detection

Deepfake videos use the idea of mimicking the movements of one video and superimposing it onto a reference video. One method of creating deepfake videos follows a similar vein of deepfake images where a GAN creates a realistic video. There are also methods called *puppet-mastering* where a person creates new movement for a target video [12]. Lip-sync videos are a form of this. In general, most deepfake video detectors use spatial-temporal information—data that is embedded in individual frames and information that is shared between frames, respectively. There is a wide variety of methods that deepfake video detectors use. Some rely on low pixel-level characteristics. Others use biometric data that deepfake tools struggle to faithfully replicate. The following sections captures the different methods that prove to be effective at detecting deepfakes.

Generally, in the pipeline of creating a deepfake, some sort of affine transformation must be made to match the resolution and face in the target video with the source video. This process can lead to discrepancies that can be distinguished in CNN models. In Li et al., a process that analyzed these discrepancies was developed with compelling results [13]. They reduced their training times as they could generate more helpful training data by employing the techniques described earlier of using Gaussian noise and Gaussian blur. Alongside that, they found that their model has the potential to be more robust than other models, as this method can be used against a broader spectrum of deepfakes. Furthermore, they found improvement to comparable models at 97.4% accuracy against the UADFV dataset, 99.9% accuracy against the low quality DeepfakeTIMIT dataset, and 93.2% accuracy against the high quality DeepfakeTIMIT dataset [13].

Another intriguing approach by Li et al. used eye movements to determine the realness of an image [14]. According to their research, deepfakes tend to have less blinking. In general, an average adult human usually blinks every 2-10 seconds with the length of a blink being within 0.1-0.4 seconds. Alongside that, the context of video, like being agitated for example, can heavily influence the frequency and length of blinking. Li et al. uses a pipeline of pre-processing and several models to detect the face, extract the features, and then process the frames of the eyes blinking through a CNN called longterm recurrent CNN (LRCN). An LRCN can consider previous temporal information to make decisions on. To combat against some attacks and improve training robustness, Li et al. trained the model with videos that had been flipped, had their contrasts increased, and had their colors and brightness distorted. They tested their approach with a LRCN, CNN, and a model called Eye Aspect Ratio (EAR). The LRCN achieved the greatest accuracy of 99%.

In [15], they develop a model named Facial Attributes-Net (FAb-Net) that can extract facial features of individuals in images. In Argawal et al., they use FAb-Net to create a temporal version that takes 4 seconds of a video and retrieves the characteristic facial data from FAb-net of each frame. Then they use another CNN that outputs a temporal encoding of the data which they called a Behavior-net. An aspect that [12] wanted to improve upon was including a quantity for identification in the detection system. FAb-net is agnostic to identity, so they also used a face recognition neural network called VGG to create an identity vector for the video. Using the vectors outputted by the Behavior-net and VGG, they do a cosine similarity on the reference data set to determine the identity of the person in the video. Finally, a video is then deemed real if the Behavior-net and VGG facial identities have the same identity, and fake otherwise. This approach tries to take combine the use of biometric data using FAb-net and low-level pixel data using VGG to determine if a video is a deepfake or not. On average, their model achieved a 94% accuracy against the datasets the used a minimum accuracy of 82.4% against the DFDC-P dataset. Furthermore, this method was also robust against compression attacks, showing that even with compressing at a lower quality, the accuracy remained nearly stagnant. Since their algorithm focuses on deepfakes that replace a video with another person, if only a subsection of the video, like a lip-sync deepfake, Argawal et al. believe that their model would not perform as well.

Ciftci et al. propose another biometric approach that uses underlying signals like heart beat, blood flow, and breathing called photoplethysmography (PPG) signals [16]. They believe that state of the art deepfake creators lack the ability to create deepfakes videos that properly account for PPG signals. Their algorithm captures PPG cells around the target's eyes and mouth, as those tend to be places that

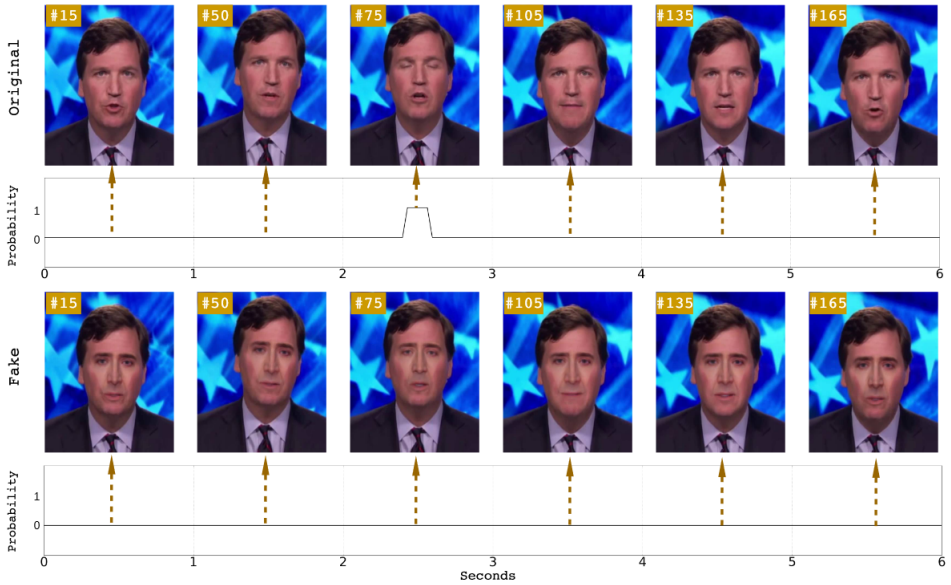

Figure 2: The discrepancy between the original and fake blink encodings [14].

exhibit these properties most obviously. They then take the PPG cell and split it up into 32 equal sized windows and find the Chrom-PPG signal for each window. They do this over ω frames of the video. The next step follows the same process, but finds the frequency domain of the PPG cells instead by calculating the power spectral density. After these two steps, the combination of the two steps results in a matrix which is processed through a simple CNN for classification. They found that using a window size of $\omega = 64$ achieves good accuracy, as a lower ω tends to not have enough valuable information and a larger ω tends to have too much noise. With this model, it could detect deepfakes with 93.69% accuracy. Unfortunately, this study does not analyze the impacts some attacks could have on the model, especially since capturing PPG data may be susceptible to video compression.

4 Conclusions and Future Research

Deepfake detection methods are a quickly growing and crucial field of research. Deepfakes pose a threat to society, so having methods to mitigate their affects is pertinent. The models presented in this paper are all manufactured in a closed environment, and they have to assume that real-world deepfakes follow similar creation methods. As a result, researchers are focusing on model independent forms of detecting deepfakes, as there is no access to the tools real-world adversaries have.

A research direction that can be further studied is integrating deepfake detection into social media platforms. Social media platforms are a means of spreading information, which make it an easy catalyst for posting deepfake content. Future research can focus on creating a preprocessing system for digital content to pass through efficiently before being uploaded.

Like in Giudice et al., another direction of research is creating more white-box models for forensics of deepfake media [7]. Most of the methods described above use black-box neural networks which give accurate but unexplainable results. The necessity of white-box models is pertinent in court of law situations where a forensics expert needs to prove the authenticity of digital media. By using a white-box method, it's clearer and persuasive as to the choice of authenticity [6].

References

[1] B. Feed, "You won't believe what obama says in this video!" Apr. 2018. [Online]. Available: https://www.youtube.com/watch?v=cQ54GDm1eL0

[2] Z. App, "Download zao app deepfake," 2019. [Online]. Available: https://zaodownload.com/download-zao-app-deepfake

[3] C. Ohman, "Introducing the pervert's dilemma: a contribution to the critique of deepfake pornography," *Ethics and Information Technology*, vol. 22, pp. 133–140, 2020. [Online]. Available: https://doi.org/10.1007/s10676-019-09522-1

[4] Aug. 2021.

[5] G. Murphy and E. Flynn, "Deepfake false memories," *Memory*, vol. 0, no. 0, pp. 1–13, 2021, pMID: 33910482. [Online]. Available: https://doi.org/10.1080/09658211.2021.1919715

[6] T. T. Nguyen, C. M. Nguyen, D. T. Nguyen, D. T. Nguyen, and S. Nahavandi, "Deep learning for deepfakes creation and detection," *CoRR*, vol. abs/1909.11573, 2019. [Online]. Available: http://arxiv.org/abs/1909.11573

[7] O. Giudice, L. Guarnera, and S. Battiato, "Fighting deepfakes by detecting GAN DCT anomalies," *CoRR*, vol. abs/2101.09781, 2021. [Online]. Available: https://arxiv.org/abs/2101.09781

[8] L. Guarnera, O. Giudice, and S. Battiato, "Fighting deepfake by exposing the convolutional traces on images," *CoRR*, vol. abs/2008.04095, 2020. [Online]. Available: https://arxiv.org/abs/2008.04095

[9] S. Agarwal, H. Farid, O. Fried, and M. Agrawala, "Detecting deep-fake videos from phoneme-viseme mismatches," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 2814–2822.

[10] X. Xuan, B. Peng, J. Dong, and W. Wang, "On the generalization of gan image forensics," *ArXiv*, vol. abs/1902.11153, 2019.

[11] Z. Guo, L. Hu, M. Xia, and G. Yang, "Blind detection of glow-based facial forgery," *Multim. Tools Appl.*, vol. 80, pp. 7687–7710, 2021.

[12] S. Agarwal, T. El-Gaaly, H. Farid, and S. Lim, "Detecting deep-fake videos from appearance and behavior," *CoRR*, vol. abs/2004.14491, 2020. [Online]. Available: https://arxiv.org/abs/2004.14491

[13] Y. Li and S. Lyu, "Exposing deepfake videos by detecting face warping artifacts," *CoRR*, vol. abs/1811.00656, 2018. [Online]. Available: http://arxiv.org/abs/1811.00656

[14] Y. Li, M. Chang, and S. Lyu, "In ictu oculi: Exposing AI generated fake face videos by detecting eye blinking," *CoRR*, vol. abs/1806.02877, 2018. [Online]. Available: http://arxiv.org/abs/1806.02877

[15] O. Wiles, A. S. Koepke, and A. Zisserman, "Self-supervised learning of a facial attribute embedding from video," *CoRR*, vol. abs/1808.06882, 2018. [Online]. Available: http://arxiv.org/abs/1808.06882

[16] U. A. Ciftci, I. Demir, and L. Yin, "How do the hearts of deep fakes beat? deep fake source detection via interpreting residuals with biological signals," *CoRR*, vol. abs/2008.11363, 2020. [Online]. Available: https://arxiv.org/abs/2008.11363