CAPSULE-FORENSICS: USING CAPSULE NETWORKS TO DETECT FORGED IMAGES AND VIDEOS

Huy H. Nguyen[⋆], *Junichi Yamagishi*[⋆†‡], *and Isao Echizen*[⋆†]

[⋆]SOKENDAI (The Graduate University for Advanced Studies), Kanagawa, Japan
[†]National Institute of Informatics, Tokyo, Japan
[‡]The University of Edinburgh, Edinburgh, UK
Email: {nhhuy, jyamagis, iechizen}@nii.ac.jp

ABSTRACT

Recent advances in media generation techniques have made it easier for attackers to create forged images and videos. State-of-the-art methods enable the real-time creation of a forged version of a single video obtained from a social network. Although numerous methods have been developed for detecting forged images and videos, they are generally targeted at certain domains and quickly become obsolete as new kinds of attacks appear. The method introduced in this paper uses a capsule network to detect various kinds of spoofs, from replay attacks using printed images or recorded videos to computer-generated videos using deep convolutional neural networks. It extends the application of capsule networks beyond their original intention to the solving of inverse graphics problems.

Index Terms— computer-generated video, replay attack, forgery detection, capsule network

1. INTRODUCTION

Forged images and videos can be used to bypass facial authentication and to create fake news media. The quality of manipulated images and videos has seen significant improvement with the development of advanced network architectures and the use of large amounts of training data. This has dramatically simplified the creation of facial forgeries. Nowadays, the only thing needed to create a forged facial image is simply a short video of the target person [1, 2] or an ID photo [3, 4]. The techniques developed by Chung et al. [4] and Suwajanakorn et al. [5] can improve the ability of attackers to learn the mapping between speech and lip motion, enabling the creation of fully synthesized audio-video data for any person. In this age of social networks serving as major sources of information, fake news with manipulated multimedia can quickly spread and have significant effects. The deepfake phenomenon [6] is a good example of this threat—any person with a personal computer can create videos incorporating the facial image of any celebrity by using a human image synthesis technique based on artificial intelligence.

Several countermeasures have been proposed to deal with manipulated images and videos. However, most of them are aimed at particular types of attacks. For example, local binary pattern (LBP)-based methods [7, 8] are effective against replay attacks in which the attacker places a printed photo or displays a video on a screen in front of the camera. However, the eyes-focused method designed to detect a deepfake forgery [9] can fail with the replay attack when the video displayed is of the actual target person. Other methods have more generalized ability; for instance, Fridrich and Kodovsky's method [10] can be applied for both steganalysis and detecting facial reenactment videos. However, its performance on secondary tasks is limited in comparison with task-specific methods like that of Rossler et al. [11]. Moreover, while some methods can detect a single forged image [11, 12, 13], others require video input [9].

This paper presents a method that uses a capsule network to detect forged images and videos in a wide range of forgery scenarios, including replay attack detection and (both fully and partially) computer-generated image/video detection. This is pioneering work in the use of capsule networks [14, 15, 16], which were originally designed for computer vision problems, to solve digital forensics problems. A comprehensive survey of state-of-the-art related work and intensive comparisons using four major datasets demonstrated the superior performance of the proposed method.

2. RELATED WORK

In this section, we group forgery detection approaches into replay attack detection and computer-generated image/video detection on the basis of the features used and their target. Note that some approaches are two-fold while others are applicable only to certain types of attacks. We also provide some basic information about capsule networks and the dynamic routing algorithm that made this kind of network practical.

978-1-5386-4658-8/18/$31.00 ©2019 IEEE

2.1. Replay Attack Detection

Prior to the current deep learning era, LBP methods were the primary defense against replay attacks [7, 8]. The method introduced by Kim et al. [17], which is based on local patterns of the diffusion speed (local speed patterns), achieves higher accuracy than that of LBP-based methods. Now, with the introduction of deep learning, the ability to detect replay attacks has been greatly improved. The method of Yang et al. [18] uses a support vector machine to classify features extracted by a pre-trained convolutional neural network (CNN). That of Menotti et al. [19] uses a similar procedure but optimizes the filters in an available high-performance CNN architecture. The method of Alotaibi and Mahmood [20] uses nonlinear diffusion based on an additive operator splitting scheme in their own CNN. The recently introduced method of Ito et al. [21] leverages a pre-trained CNN and utilizes the whole image instead of only the extracted face region.

2.2. Computer-Generated Image/Video Detection

There are several state-of-the-art methods for detecting images or videos generated by computer using, for example, a deepfake technique for face swapping [6], the Face2Face method for facial reenactment [1], or the deep video portraits technique [2] for the purpose of forgery. Fridrich and Kodovsky [10] proposed a hand-crafted-feature noise-based approach for steganalysis that can also be used for forgery detection. Cozzolino et al. [22] implemented a CNN version of this approach. Raghavendra et al. [23] described the special case of fine-tuning two available CNNs while Rossler et al. [11] used only one CNN. Bayar and Stamm [24], Rahmouni et al. [25], Afchar et al. [13], Quan et al. [26], and Li et al. [9] proposed their own networks. Li et al.'s network [9], for example, is video based and uses temporal information to detect eye blinking. We used a hybrid approach [12] incorporating part of a pre-trained VGG (Visual Geometry Group)-19 network [27] and a proposed CNN. Zhou et al. [28] proposed a two-stream network.

2.3. Capsule Networks

Hinton et al. [14] addressed the limitations of CNNs applied to inverse graphics tasks and laid the foundation for a more robust "capsule" architecture in 2011. However, this complex architecture could not be effectively implemented at the time due to the lack of an efficient algorithm and the limitations of computer hardware. Instead, easy-to-design easy-to-train CNNs became widely used. Now, with the introduction of the dynamic routing algorithm [15] and the expectation-maximization routing algorithm [16], capsule networks have been implemented with remarkable initial results. Two recent studies demonstrated that, with the agreement between capsules calculated by the dynamic routing algorithm, the hierarchical pose relationships between object parts can be well described. This has improved the accuracy of vision tasks. Application of a capsule network to the forensics task, the focus of this paper, is a challenging problem. However, the agreement between capsules achieved by using the dynamic routing algorithm could boost detection performance on complex and nearly flawless forged images and videos.

3. CAPSULE-FORENSICS

3.1. Overview

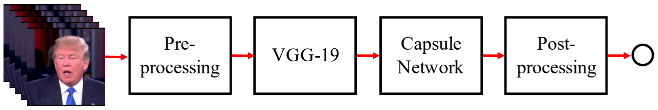

Fig. 1. Overview of proposed method.

The proposed method (Fig. 1) works for both images and videos. For video input, the video is split into frames in the pre-processing phase. The classification results (posterior probabilities) are then acquired from the frames. The probabilities are averaged in the post-processing phase to get the final result. The remaining parts are constructed the same way as when the input is an image.

In the pre-processing phase, faces are detected and scaled to 128 × 128. Like we did in our previous work [12], we use part of the VGG-19 network [27] to extract the latent features, which are the inputs to the capsule network. Unlike we did in our previous work, we take the output of the third maxpooling layer instead of three outputs before the ReLU layers. We do this because we need to reduce the size of the inputs to the capsule network.

3.2. Capsule Design

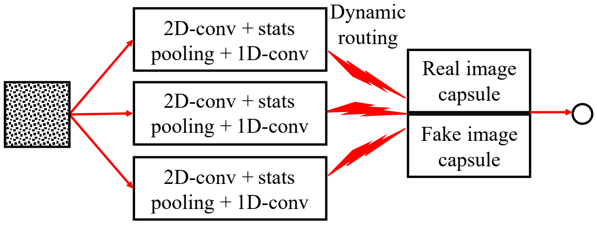
2D-conv + stats pooling + 1D-conv
2D-conv + stats pooling + 1D-conv
2D-conv + stats pooling + 1D-conv
Dynamic routing
Real image capsule
Fake image capsule

Fig. 2. Overall design of capsule-forensics network.

The proposed network consists of three primary capsules and two output capsules, one for real and one for fake images (Fig. 2). The latent features extracted by part of the VGG-19 network [27] are the inputs, which are distributed to the three primary capsules (Fig. 3). As in our previous work [12], statistical pooling, which is important for forgery detection, is used. The outputs of the three capsules ($\mathbf{u}_{j|i}$) are dynamically

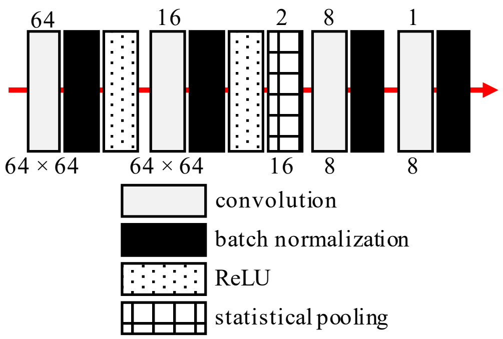

Fig. 3. Detailed design of primary capsule. Upper numbers indicate number of filters (depth) while lower number indicate size of outputs of corresponding filters.

Algorithm 1 Dynamic routing between capsules.

```
procedure ROUTING($\mathbf{u}_{j|i}, W, r$)
    $\hat{W} \leftarrow W + rand(size(W))$
    $\hat{\mathbf{u}}_{j|i} \leftarrow \hat{W}_i squash(\mathbf{u}_{j|i})$        $\triangleright W_i \in R^{m \times n}$
    for all input capsule $i$ and all output capsules $j$ do
        $b_{ij} \leftarrow 0$
    for $r$ iterations do
        for all input capsules $i$ do $c_i \leftarrow softmax(b_i)$
        for all output capsules $j$ do $\mathbf{s}_j \leftarrow \sum_i c_{ij} \hat{\mathbf{u}}_{j|i}$
        for all output capsules $j$ do $\mathbf{v}_j \leftarrow squash(\mathbf{s}_j)$
        for all input capsules $i$ and output capsules $j$ do
            $b_{ij} \leftarrow b_{ij} + \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$
    return $\mathbf{v}_j$
```

routed to the output capsules ($\mathbf{v}_j$) for r iterations using Algorithm 1. The network has approximate 2.8 million parameters, a relatively small number for such networks. We slightly improved the algorithm of Sabour et al. [15] by adding Gaussian random noise to the 3-D weight tensor W and applying one additional $squash$ (equation 1) before routing by iterating. The added noise helps reduce over-fitting while the additional equation keeps the network more stable. The outputs of the primary and output capsules are illustrated in Fig. 4.

$$\mathbf{v}_j = squash(\mathbf{s}_j) = \frac{\|\mathbf{s}_j\|^2}{1 + \|\mathbf{s}_j\|^2} \frac{\mathbf{s}_j}{\|\mathbf{s}_j\|} \tag{1}$$

Unlike Sabour et al.'s work [15], we use the cross-entropy loss function:

$$L = -\left(y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})\right), \tag{2}$$

where y is the ground truth label and $\hat{y}$ is the predicted label calculated using equation 3, in which m is the dimension of the output capsule $\mathbf{v}_j$.

$$\hat{y} = \frac{1}{m} \sum_i softmax\left(\begin{bmatrix}\mathbf{v}_1^\intercal \\ \mathbf{v}_2^\intercal\end{bmatrix}_{:,i}\right) \tag{3}$$

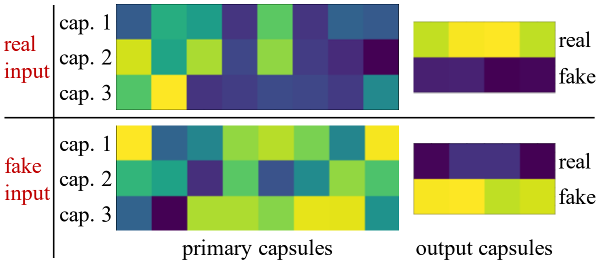

Fig. 4. Average results calculated by primary capsules and output capsules from real and fake images generated with Face2Face method [1]. Three primary capsules have significantly different reactions between real and fake inputs. Although their weights are also different, there is strong agreement in the output capsules.

The use of equation 3 instead of simply using the length of the output capsules [15] promotes separation between the two output capsules on each dimension.

4. EVALUATION

To evaluate the advantage of using random noise, we tested the proposed method with and without using random noise (Capsule-Forensics-Noise and Capsule-Forensics, respectively). The random noise was generated from a normal distribution $N(0, 0.01)$ and was used in the training phase only. Two iterations ($r = 2$) were used in the dynamic routing algorithm. We used the half total error rate (HTER) $\left(\frac{FRR+FAR}{2}\right)$ and accuracy $\left(\frac{TP+TN}{TP+TN+FP+FN}\right)$ as metrics.

4.1. Replay Attack Detection

To determine the ability of the proposed method to detect replay attacks, we compared its performance with that of eight state-of-the-art detection methods on the well-known Idiap REPLAY-ATTACK dataset [7]. As shown in Table 1, the proposed method with random noise (Capsule-Forensics-Noise), as well as our previous method [12], had an HTER of zero.

Table 1. Half total error rate of state-of-the-art detection methods on REPLAY-ATTACK dataset [7].

Method	HTER (%)
Chigovska et al. [7]	17.17
Pereira et al. [8]	08.51
Kim et al. [17]	12.50
Yang et al. [18]	02.30
Menotti et al. [19]	00.75
Alotabib et al. [20]	10.00
Ito et al. [21]	00.43
Nguyen et al. [12]	**00.00**
Capsule-Forensics	00.28
Capsule-Forensics-Noise	**00.00**

4.2. Face Swapping Detection

We determined the ability of our proposed method to detect face swapping using a deepfake technique on the deepfake dataset proposed by Afchar et al. [13] at both the frame and video levels. As shown in Tables 2 and 3, our proposed method with random noise (Capsule-Forensics-Noise) had the highest accuracy in both cases.

Table 2. Accuracy of face swapping detection at frame level on deepfake dataset [13].

Method	Accuracy (%)
Meso-4 [13]	89.10
MesoInception-4 [13]	91.70
Nguyen et al. [12]	92.36
Capsule-Forensics	94.47
Capsule-Forensics-Noise	**95.93**

Table 3. Accuracy of face swapping detection at video level on deepfake dataset [13].

Method	Accuracy (%)
Meso-4 [13]	96.90
MesoInception-4 [13]	98.40
Capsule-Forensics	97.69
Capsule-Forensics-Noise	**99.23**

4.3. Facial Reenactment Detection

We determined the ability of our proposed method to detect facial reenactment on the FaceForensics dataset [11], which was created using the Face2Face method [1]. We strictly followed the authors' guidelines for processing the data. As shown in Table 4, on average, the proposed method (with and without noise) had performance comparable to that of the best-performing state-of-the-art methods.

We also tested our method at the video level and compared its performance with that of Afchar et al.'s MesoNet facial video forgery detection network [13]. For our method, we used only the first ten frames instead of the entire video. As shown in Table 5, our method outperformed Afchar et al.'s network.

4.4. Fully Computer-Generated Image Detection

Finally, we compared the performance of our proposed method with that of state-of-the-art methods on computer-generated images (CGIs) and photographic images (PIs) on the dataset proposed by Rahmouni et al. [25]. Once again, as shown in Table 6, our method had the best performance and had perfect accuracy on full-size test images.

5. CONCLUSION

Our comprehensive experiments demonstrated the feasibility of building a general detection method that is effective for a wide range of forged image and video attacks. They also demonstrated that capsule networks can be used in domains other than computer vision. The proposed use of random noise in the training phase proved beneficial in most cases. Future work will mainly focus on evaluating the ability of the proposed method to resist adversarial machine attacks, especially on the proposed random noise at test time, and enhancing its ability. It will also focus on making the proposed method robust against mixed attacks, on detecting anomalies, and on raising this critical issue in the research community.

Table 4. Accuracy of state-of-the-art facial reenactment detection methods at frame level on FaceForensics dataset [11] with three levels of compression: no compression, easy compression (23), and strong compression (40).

Method	Accuracy (%)		
	No-C	Easy-C	Hard-C
Fridrich & Kodovsky [10]	99.40	75.87	58.16
Cozzolino et al. [22]	99.60	79.80	55.77
Bayar & Stamm [24]	99.53	86.10	73.63
Rahmouni et al. [25]	98.60	88.50	61.50
Raghavendra et al. [23]	97.70	93.50	82.13
Zhou et al. [28]	99.93	96.00	86.83
Rossler et al. [11]	99.93	98.13	87.81
Meso-4 [13]	94.60	92.40	83.20
MesoInception-4 [13]	96.80	93.40	81.30
Nguyen et al. [12]	98.80	96.10	76.40
Capsule-Forensics	99.13	97.13	81.20
Capsule-Forensics-Noise	99.37	96.50	81.00

Table 5. Comparison with MesoNet network at video level on FaceForensics dataset [11].

	Accuracy (%)		
	No-C	Easy-C	Hard-C
Meso-4 [13]	-	95.30	-
MesoInception-4 [13]	-	95.30	-
Capsule-Forensics	**99.33**	**98.00**	82.00
Capsule-Forensics-Noise	**99.33**	96.00	**83.33**

Table 6. Accuracy of state-of-the-art methods on discriminating between CGIs and PIs.

Method	Accuracy	
	Patch	Full Size
Rahmouni et al. [25]	89.76	99.30
Quan et al. [26]	94.75	99.58
Nguyen et al. [12]	96.55	99.86
Capsule-Forensics	96.75	99.72
Capsule-Forensics-Noise	**97.00**	**100.00**

6. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Numbers (16H06302, 17H04687, 18H04120, 18H04112, 18KT0051) and by JST CREST Grant Number JPMJCR18A6, Japan.

7. PREFERENCES

[1] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner, "Face2Face: Real-time face capture and reenactment of RGB videos," in *CVPR*. IEEE, 2016.

[2] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Nießner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt, "Deep video portraits," in *SIGGRAPH*. ACM, 2018.

[3] Hadar Averbuch-Elor, Daniel Cohen-Or, Johannes Kopf, and Michael F Cohen, "Bringing portraits to life," *ACM TOG*, 2017.

[4] Joon Son Chung, Amir Jamaludin, and Andrew Zisserman, "You said that?," *arXiv preprint arXiv:1705.02966*, 2017.

[5] Supasorn Suwajanakorn, Steven M Seitz, and Ira Kemelmacher-Shlizerman, "Synthesizing obama: learning lip sync from audio," *ACM TOG*, 2017.

[6] "Terrifying high-tech porn: Creepy 'deepfake' videos are on the rise," https://www.foxnews.com/tech/terrifying-high-tech-porn-creepy-deepfake-videos-are-on-the-rise, Accessed: 2018-02-17.

[7] Ivana Chingovska, André Anjos, and Sébastien Marcel, "On the effectiveness of local binary patterns in face anti-spoofing," in *BIOSIG*, 2012.

[8] Tiago de Freitas Pereira, André Anjos, José Mario De Martino, and Sébastien Marcel, "Can face anti-spoofing countermeasures work in a real world scenario?," in *ICB*. IEEE, 2013.

[9] Yuezun Li, Ming-Ching Chang, Hany Farid, and Siwei Lyu, "In ictu oculi: Exposing AI generated fake face videos by detecting eye blinking," *arXiv preprint arXiv:1806.02877*, 2018.

[10] Jessica Fridrich and Jan Kodovsky, "Rich models for steganalysis of digital images," *IEEE TIFS*, 2012.

[11] Andreas Rössler, Davide Cozzolino, Luisa Verdoliva, Christian Riess, Justus Thies, and Matthias Nießner, "FaceForensics: A large-scale video dataset for forgery detection in human faces," *arXiv preprint arXiv:1803.09179*, 2018.

[12] Huy H Nguyen, Ngoc-Dung T Tieu, Hoang-Quoc Nguyen-Son, Vincent Nozick, Junichi Yamagishi, and Isao Echizen, "Modular convolutional neural network for discriminating between computer-generated images and photographic images," in *ARES*. ACM, 2018.

[13] Darius Afchar, Vincent Nozick, Junichi Yamagishi, and Isao Echizen, "MesoNet: a compact facial video forgery detection network," in *WIFS*. IEEE, 2018.

[14] Geoffrey E Hinton, Alex Krizhevsky, and Sida D Wang, "Transforming auto-encoders," in *ICANN*. Springer, 2011.

[15] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton, "Dynamic routing between capsules," in *NIPS*, 2017.

[16] Geoffrey E Hinton, Sara Sabour, and Nicholas Frosst, "Matrix capsules with EM routing," in *ICLRW*, 2018.

[17] Wonjun Kim, Sungjoo Suh, and Jae-Joon Han, "Face liveness detection from a single image via diffusion speed model," *IEEE TIP*, 2015.

[18] Jianwei Yang, Zhen Lei, and Stan Z Li, "Learn convolutional neural network for face anti-spoofing," *arXiv preprint arXiv:1408.5601*, 2014.

[19] David Menotti, Giovani Chiachia, Allan Pinto, William Robson Schwartz, Helio Pedrini, Alexandre Xavier Falcao, and Anderson Rocha, "Deep representations for iris, face, and fingerprint spoofing detection," *IEEE TIFS*, 2015.

[20] Aziz Alotaibi and Ausif Mahmood, "Deep face liveness detection based on nonlinear diffusion using convolution neural network," *Signal, Image and Video Processing*, 2017.

[21] Koichi Ito, Takehisa Okano, and Takafumi Aoki, "Recent advances in biometrics security: A case study of liveness detection in face recognition," in *APSIPA ASC*. IEEE, 2017.

[22] Davide Cozzolino, Giovanni Poggi, and Luisa Verdoliva, "Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection," in *IH&MMSEC*. ACM, 2017.

[23] R. Raghavendra, Kiran B. Raja, Sushma Venkatesh, and Christoph Busch, "Transferable deep-CNN features for detecting digital and print-scanned morphed face images," in *CVPRW*. IEEE, 2017.

[24] Belhassen Bayar and Matthew C Stamm, "A deep learning approach to universal image manipulation detection using a new convolutional layer," in *IH&MMSEC*. ACM, 2016.

[25] Nicolas Rahmouni, Vincent Nozick, Junichi Yamagishi, and Isao Echizen, "Distinguishing computer graphics from natural images using convolution neural networks," in *WIFS*. IEEE, 2017.

[26] Weize Quan, Kai Wang, Dong-Ming Yan, and Xiaopeng Zhang, "Distinguishing between natural and computer-generated images using convolutional neural networks," *IEEE TIFS*, 2018.

[27] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[28] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis, "Two-stream neural networks for tampered face detection," in *CVPRW*. IEEE, 2017.