Forensic Science International: Digital Investigation 36 (2021) 301108

Contents lists available at ScienceDirect

Forensic Science International: Digital Investigation

journal homepage: www.elsevier.com/locate/fsidi

Learning Spatio-temporal features to detect manipulated facial videos created by the Deepfake techniques

Xuan Hau Nguyen [a], Thai Son Tran [a], Van Thinh Le [b], Kim Duy Nguyen [c], Dinh-Tu Truong [d, e, *]

[a] *Mientrung University of Civil Engineering, Viet Nam*
[b] *Faculty of Information Technology, Central Industrial and Commercial College, Viet Nam*
[c] *Institute of Engineering-Technology, Thu Dau Mot University, Viet Nam*
[d] *Natural Language Processing and Knowledge Discovery Laboratory, Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Viet Nam*
[e] *Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Viet Nam*

ARTICLE INFO

Article history:
Received 24 June 2020
Received in revised form
5 January 2021
Accepted 5 January 2021
Available online 20 January 2021

Keywords:
Video forensic
Deepfake video detection
Autoencoder-decoder
Generative adversarial networks
Deep learning
3-D convolutional neural network

ABSTRACT

In the last years, the face synthetic video generation has been rapid, and hyper-realistic forged videos are based on Deepfake techniques. It leads to a loss of trust in videos' content and makes it malicious by spreading forged videos on the internet. Until now, there are a few algorithms that have been suggested for detecting forged videos created by Deepfake techniques, but most of them based on analyzing or learning features on frames separately in a video. Those algorithms often pay less attention to Spatio-temporal features, so these algorithms' accuracy is usually not good. This paper proposes a 3-dimensional (3-D) convolutional neural network model that can learn Spatio-temporal features from an adjacent frame sequence in a video. Our proposed network's binary detection accuracy reached over 99% on the two largest benchmark datasets as Deepfake of FaceForensics++ and VidTIMIT datasets. The experimental results of the proposed method outperform state-of-the-art methods.

I. Introduction

Nowadays, the popularization of digital tools and the development of social networks have made digital videos very ordinary objects. Every day, millions of videos are uploaded on the internet. Most of them have been manipulated by techniques to change video content. And the spread of forged videos has raised dangerous consequences for individuals and society. Notably, in the last two years, Deep learning-based face replacement tools have been rapidly developed in the video. These critical tools are Faceswap (2019), Faceswap-GAN (Faceswap-GAN, 2019), DeepFaceLab (DeepFaceLab, 2019), and DFaker (Dfaker, 2019) used to make videos in which contain face tampering. The naked eyes hardly distinguish those facial video forgeries. They can be made for malicious purposes as pornographic videos of celebrities, politicians, fake news, fake surveillance videos, and policy tensions. So, nowadays, facial video forgery detection has become a hot topic of interest amongst researchers worldwide. And in December 2019 year, Facebook, Microsoft, some largest universities and partners have organized the Deepfake Detection Challenge designed to incentivize rapid progress in this area by inviting participants to create new ways of detecting and preventing manipulated media content.

Deepfake was defined by combining deep learning and fake, which is the most powerful technique to create forged multimedia content. Deepfake videos were faked videos created by Deepfake tools. Through Deepfake tools, the target person's face is transferred to a video of a source person to create a new video in that it has the face of the target person but action as the source person, as shown in Fig. 1. Today with powerful hardware and deep learning models based on autoencoders or generative adversarial network (GAN). It is fully automatic to create realistic fake videos. Deepfake tools such as Faceswap, Faceswap-GAN, etc., can be used to produce a realistic fake video by anyone who is only basically trained. Most

* Corresponding author. Ton Duc Thang University, Ho Chi Minh City, Viet Nam.
E-mail addresses: nguyenxuanhau@tic.edu.vn (X.H. Nguyen), tranthaison@muce.edu.vn (T.S. Tran), levanthinh@tic.edu.vn (V.T. Le), duynk@tdmu.edu.vn (K.D. Nguyen), truongdinhtu@tdtu.edu.vn (D.-T. Truong).

https://doi.org/10.1016/j.fsidi.2021.301108
2666-2817/© 2021 Elsevier Ltd. All rights reserved.

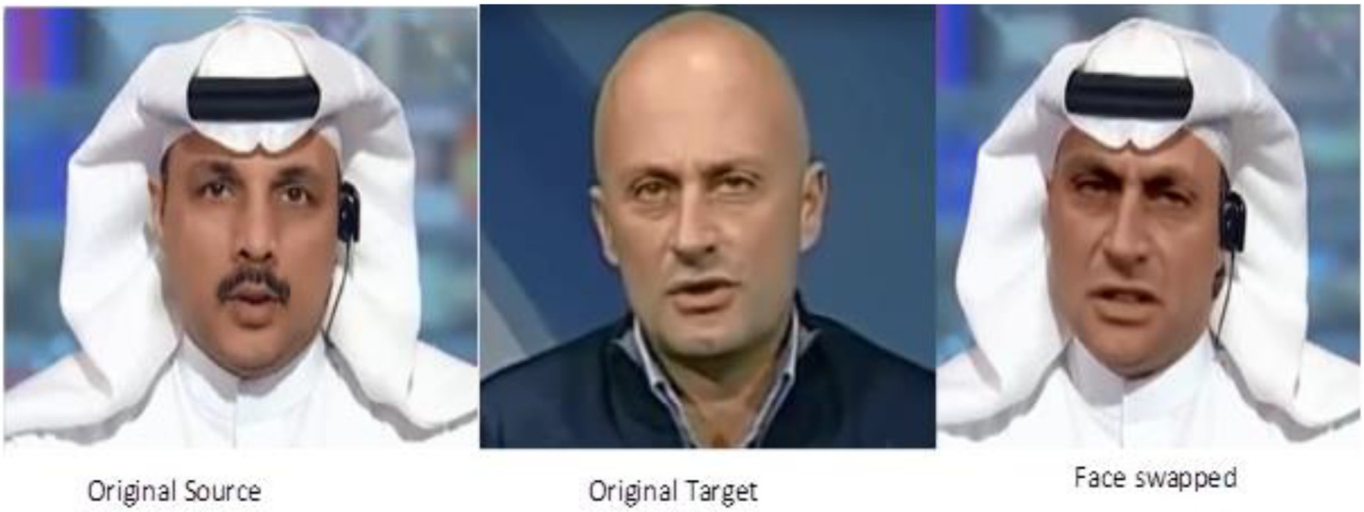


Fig. 1. A Deepfake example, the picture got from the FaceForensics++ dataset.

of the actions throw Graphical User Interfaces (GUI) without effort. Deepfake videos are very malicious when they are fake celebrity pornography videos, fake politician videos. It is not only a terrible impact on an individual but also the community (Dolhansky et al., 2019).

Recently, some proposed methods detected facial tampered video. Most of which are only based on steganalysis of handcrafted-features or convolutional neural networks (CNN) on frames separately. They have not exploited features that have relation in spatial and temporal between adjacent frames in the video. Therefore, all of the recently proposed methods have not given good results, and the detection of Deepfake videos is still a significant challenge. The state-of-the-art machine learning techniques showed that 3-dimensional (3-D) convolution kernels could learn spatial and temporal features simultaneously and achieved breakthrough Performance (Tran et al., 2015). Therefore, in this research, we have applied and have proposed using 3-D convolution kernels to build a deep 3-D convolutional neural network (CNN) to learn Spatio-temporal features in short consecutive frames sequence to detect Deepfake videos. We have experimented on the two largest and popular Deepfake video datasets as FaceForensics++ (Rössler et al., 2019) and VidTIMIT (Korshunov and Marcel, 2018) and compared the proposed methods' efficiency with the-state of-the-art methods. Through that, it proved our proposed method is more efficient and accurate than modern methods. The detail of the proposed method is in section III.

To summarize, this paper makes the following contributions:

- We have applied the 3-D convolution kernels to build a deep 3-D convolutional neural network that extracts Spatio-temporal features from short consecutive frames sequence for detecting Deepfake videos.
- We have also proposed a method to extract faces in consecutive frames sequence in a video to construct 3D images, which are the proposed model's input.
- We have experimented on the two largest Deepfake datasets. The results show that the proposed model is good to learn features in spatial and temporal.
- The proposed network is a simple, effective approach, and it is easy to implement to use.

The rest of the paper is organized as follows. In section II is related works. In section III, we present the proposed method. Experiments are given in section IV, and section V contains conclusions and future directions.

2. Related work

Recently, Deepfake videos have a strong negative effect on privacy, social security, and democracy (Sabir et al., 2019). So, there are some methods for detecting Deepfake videos that have been proposed. They can be divided into two groups as temporal features based methods and spatial features based methods.

2.1. Temporal features-based deep learning methods

Most of the methods in this group have based on the observation that an original video is a consecutive frame sequence, and pixels' value in a video has significantly coherent in Spatio-temporal. Otherwise, a Deepfake video was created from processing facial synthesis by a frame privately. So, pixels' values in synthesis areas are not consistent in Spatio-temporal. There are typical proposed methods as follows: Sabir et al. (2019) used a convolutional neural network (CNN) to extract features from the face area of the frame separately. Those features were later inputted into the recurrent convolutional network (RNN) to exploit temporal information from consecutive frames. Likewise, Guera et al. (Güera and Delp, 2018) showed that there are intra-inconsistencies and temporal inconsistencies between frames. The authors combined CNN for extracting features from frames and long short term memory (LSTM) network for capturing temporal inconsistencies between frames. On the other hand, Li et al. (2018) used eye blinking signals to distinguish between Deepfake videos and original ones based on the observation that in a Deepfake video, a person's eye is less frequent blinking than that in an original video. The bounding boxes of eye landmark point sequences were created after a few pre-processing steps, such as aligning face, extracting, and scaling. They were inputted into the long-term recurrent convolutional networks for distinguishing eye blinking sequences between tampered videos and original videos. Most of the methods in this group are based on deep recurrent networks, which learn temporal patterns across video frames.

2.2. Spatial features-based deep learning methods

The difference with temporal features based methods, methods in this group explore artifact patterns inside single frames separately in videos. These methods are based on the observation that Deepfake videos normally were created by an affine face wrapping approach to match original faces in the source videos on the frame privacy, which creates inconsistent between the warped face area and the surrounding context. There are typical proposed methods as follows: Li et al. (Li and Lyu, 2018) used the power models in deep learning such as VGG and ResNet to capture those inconsistent features at frame privacy in Deepfake videos. Similarly, in (Afchar et al., 2018), the authors proposed two deep learning models with a low number of layers to distinguish frames between Deepfake videos and original videos. This approach has experimented with a dataset with high compressed videos, but the result is not so good. In (Nguyen et al., 2019), the authors applied a capsule network to get features from the VGG-19 network to distinguish the Deepfake videos from the original ones.

2.3. Spatial features based shallow classifiers

Some methods used a physiological signal for classifying Deepfake and the original face. Such as in (Yang et al., 2019), the authors estimated 3-D head poses from the face area, and it is based on 68 facial landmarks of the central face region. The SVM classifier is then used to classify Deepfake and the original face. Similarly, in (Matern et al., 2019), the authors used features such as eyes, teeth, and facial contours for detecting Deepfake videos. Besides that, the photoresponse non-uniformity (PRNU) is also used in (Koopman et al., 2018) to distinguish the Deepfake video from the original one.

Up to now, all of the approaches above only analyzed either temporary or spatial features. So, with the purpose increase the efficiency of detecting Deepfake videos, we have built a 3-D convolutional deep network that extracts Spatio-temporal features efficiently for detecting Deepfake videos.

3. Proposed method

1. Problem formulation

Deepfake videos have gained popularity because of the high quality of fake videos and the ease of using tools to create them. Even novices can create Deepfake videos. The tools creating Deepfake videos are mostly developed based on deep learning techniques.

Deep learning is perfect, with the ability to represent multidimensional and highly complex data. A specific case of the deep learning model with the ability mentioned above is autoencoders, which are widely applied to dimensional reduction and image data compression (Chorowski et al., 2019; Cheng, 2019; Punnappurath and Brown, 2019). The first tool to create Deepfake videos is FakeApp, developed by a Reddit user using an autoencoder-decoder network structure (FakeApp 2.2.0; Faceswap). The autoencoder extracts the latent features of faces in those methods, and the decoder is used to rebuild faces. Swap faces between source and target images; two encoder-decoder pairs are needed. Each pair is used to learn on a different set of images, and the variables of the encoders are shared between the two pairs network.

The strategy above allows the encoder to share and learn the similarities between the two sets of face images because the faces have the same features as the eyes, nose, mouth. Fig. 2 shows the process of creating a Deepfake, in which the features of the original source face are connected to the decoder of the original target face to reconstruct the original target face from the original source. This approach is applied to create the Deepfake video and typical tools for creating Deepfake videos like DeepFake-tf, DeepFaceLab, and Dfaker. Besides, to improve Deepfake video quality, some tools are based on generative adversarial networks (GAN) like the faceswap-GAN tool proposed recently.

3.1. Proposed method

Because Deepfake videos are created by manipulating each frame separately, these separate frames are rendered into Deepfake videos. So the facial synthesis in the Deepfake videos not only has inconsistent information on spatial but also on the temporal dimension. To distinguish between Deepfake and original videos, we have proposed a method based on 3-D CNN by learning information in both spatial and temporal dimensions from consecutive faces that have been gotten from consecutive frame sequence.

We have applied the 3-D convolution kernels to build deep 3-D convolutional networks that extract Spatio-temporal features from a short consecutive frame sequence for detecting Deepfake videos. The 3-D convolution is done by multiplying a 3-D kernel with a cube that is stacked successive frames. In this implementation, the feature maps in the convolutional layers are connected to the previous layer's successive frames. So it can capture information on the faces in consecutive frames sequence at spatial and temporal dimensions.

We have started experiments with complex network models and incorporating different kernel sizes (kernel size as $7 \times 7 \times 3$, $5 \times 5 \times 3$, and $3 \times 3 \times 3$). They are gradually simplified until we get the following model, but the results are not still inferior, and the number of network parameters and computations are the least.

We have also performed experiments with the different inputs as 32, 24, 16, 8, and 4 consecutive faces on the proposed networks with the approximately parameter number. These consecutive faces are extracted from a consecutive frame sequence in video. The result with the input as 16 consecutive faces is best. The proposed network architecture is in Fig. 3. This network's input is a consecutive 16 faces sequence extracted from a consecutive 16 frames sequence in the video. This network begins with a sequence of five layers set of successive convolution, batch normalization, ReLU, and Max pooling. The feature maps number increase from 8 to 128 from the first layer to the fifth layer. The spatial dimension decreases a half through each layer, decreasing from 128 to 4 dimensions from the first layer to the fifth layer. And especially, the temporal dimension does not change at the first layer, but they only decrease half from the second layer to the fifth layer (from 16 dimensions to 1 dimension). Generalization enhanced the above layers using the ReLU functions, and in addition, to normalize the output and prevent vanishing gradient effect, those layers also use Batch Normalizations (Ioffe and Szegedy, 2015). Following the five layers above are a Dropout layer (Srivastava et al., 2014) and two fully-connected layers to regularize and improve the model's effectiveness. In total, this network architecture has about 1.3 million parameters; more details can be found in Fig. 3.

4. Experiments

In this section, we present how to prepare data from available datasets, the datasets for experiments are Deepfake videos of FaceForensics ++ (Rössler et al., 2019) and VidTIMIT (Korshunov and Marcel, 2018) datasets. Later, the empirical results of the proposed model are presented. Besides, we have also compared the results with some latest research on the same datasets above.

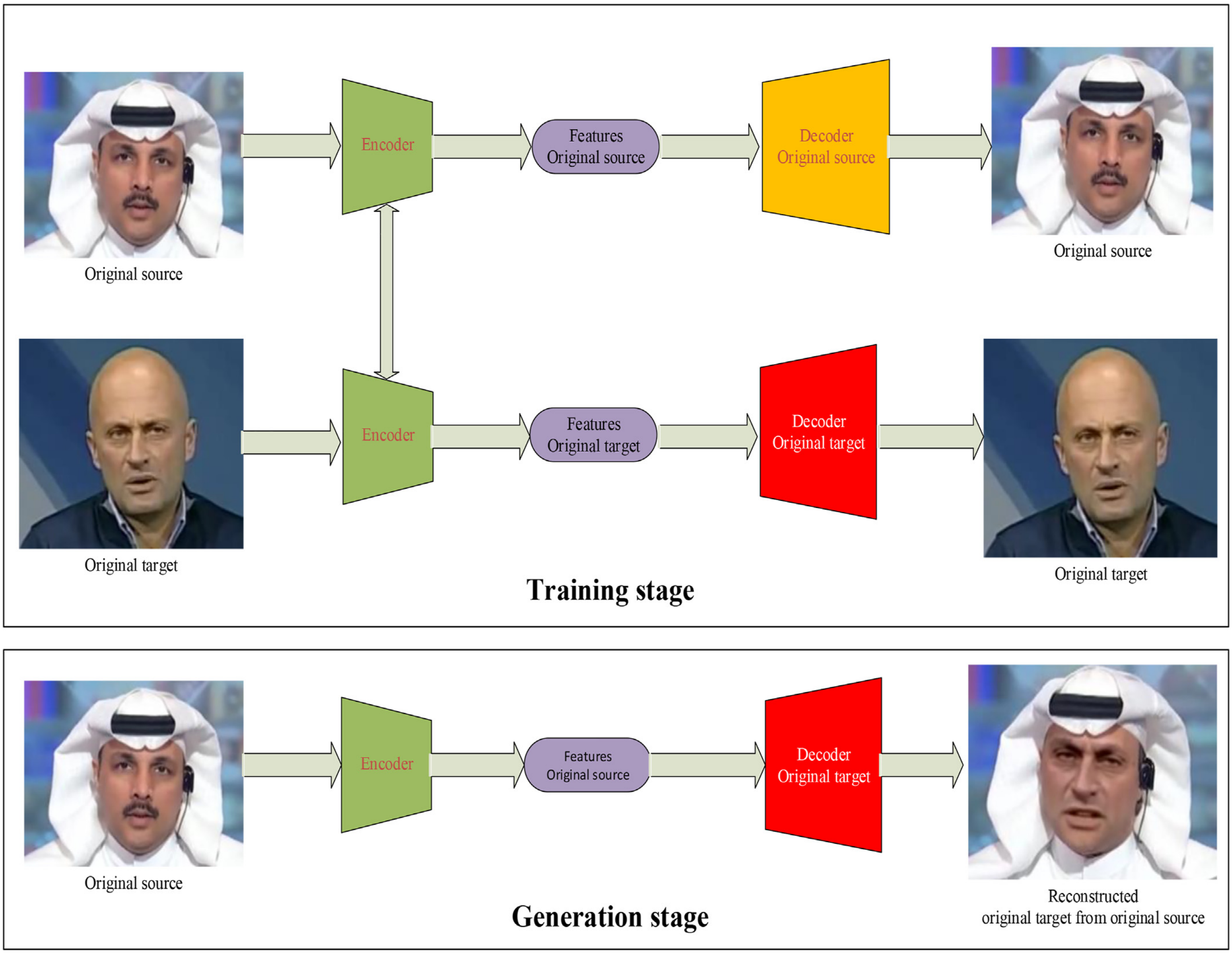


Fig. 2. A model creates Deepfakes. In the training stage, two networks use the same encoder but different decoders for the training stage. Then in the generation stage, the original source is encoded with the common encoder and decoded with the decoder of the original target to create a Deepfake.

4.1. Data preparation

FaceFornsics ++ dataset has two parts: high-quality and low-quality parts. Each part was split into three fixed subsets as the training set (720 original and 720 Deepfake videos), the validation set (140 original and 140 Deepfake videos), and the test set (140 original and 140 Deepfake videos). And with the VidTIMIT dataset, the number of Deepfake videos is not much, with only 320 Deepfake videos in each high-quality and low-quality parts. Therefore, each part was split into two fixed subsets as the training set (309 original and 240 Deepfake videos) and the test set (80 original and 80 Deepfake videos).

The input of the proposed model is a 3-D image with a size of (128,128,16,3). Therefore, the 3-D image of face regions on a consecutive 16 frames sequence would be extracted. The proposed model would be trained to learn consistent or inconsistent information of the face region in both spatial and temporal dimensions on the 3-D images. We have also proposed constructing a 3-D image by extracting face regions from 16 consecutive frames as follows: we would check the location of one face region on consecutive 16 frames does not change a lot. The face regions on each frame were extracted at the same spatial location to construct the 3-D image. The detail of obtaining face region procession into the 3-D image is shown in the appendix.

The data preparation was performed on Deepfake videos of FaceForensic++ and VidTIMIT datasets. Finally, we have gotten two datasets for training. The summary of the training dataset has been shown in Table 1.

4.2. Model configuration for training

We have set training parameters as follows: We used the SGD optimization method with momentum-contribution of the previous step is 0.9. The initial learning rate set is 0.001. The learning rate would drop to 0.1 after ten epochs. Mini_batch_size is 20, Max_epochs is 30, shuffle at every epoch, and L2 regularization is 0.0001. Figs. 4 and 5 show the progressing of the training model on high-quality and low-quality parts of FaceForensic++. In Fig. 4, when we train the proposed model on the high-quality part, validation accuracy is 99.33%. In Fig. 5, when we train the proposed model on the low-quality part, validation accuracy is 94.85%.

4.3. Test results

For testing, each video in the testing dataset part would be followed by guides in Data preparation for creating 3-D images that

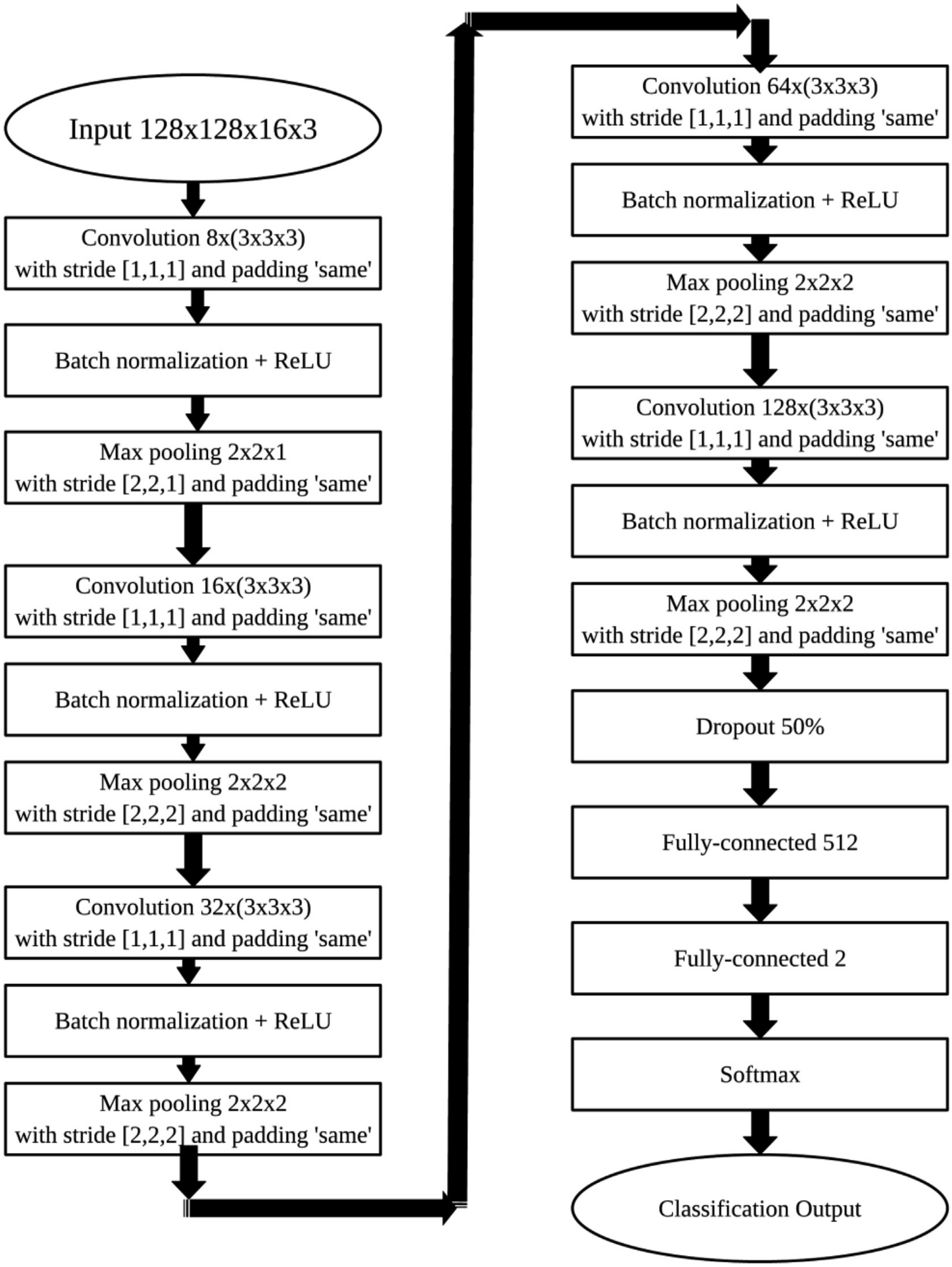
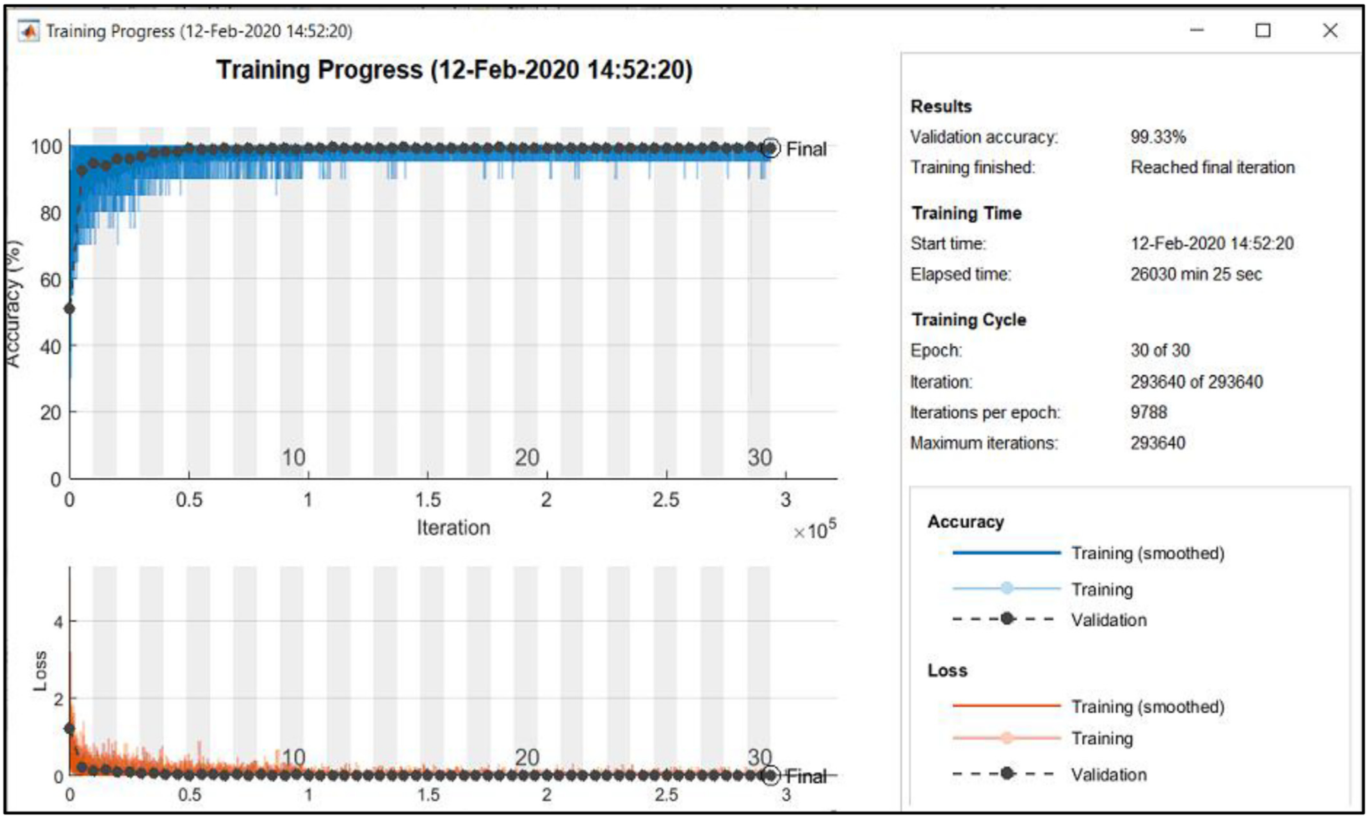

Input 128x128x16x3
Convolution 8x(3x3x3)
with stride [1,1,1] and padding 'same'
Batch normalization + ReLU
Max pooling 2x2x1
with stride [2,2,1] and padding 'same'
Convolution 16x(3x3x3)
with stride [1,1,1] and padding 'same'
Batch normalization + ReLU
Max pooling 2x2x2
with stride [2,2,2] and padding 'same'
Convolution 32x(3x3x3)
with stride [1,1,1] and padding 'same'
Batch normalization + ReLU
Max pooling 2x2x2
with stride [2,2,2] and padding 'same'
Convolution 64x(3x3x3)
with stride [1,1,1] and padding 'same'
Batch normalization + ReLU
Max pooling 2x2x2
with stride [2,2,2] and padding 'same'
Convolution 128x(3x3x3)
with stride [1,1,1] and padding 'same'
Batch normalization + ReLU
Max pooling 2x2x2
with stride [2,2,2] and padding 'same'
Dropout 50%
Fully-connected 512
Fully-connected 2
Softmax
Classification Output

Fig. 3. The 3-D network architecture for detecting Deepfake.

Table 1
The summary of the Deepfake training datasets of FaceForensic++ and VidTIMIT.

FaceForensics++		VidTIMIT
Number of videos	740 original and 740 Deepfake videos	309 original and 240 Deepfake videos
Number of the 3-D images as negative samples extracted from original videos	100,658	9567
Number of the 3-D images as positive samples extracted from Deepfake videos	98,754	8231

are considered samples. From that, we would have a set of samples from each video. That set of samples would be classified by the trained models above. The output of the model classifies the samples are forged or original. Besides, to demonstrate the



Training Progress (12-Feb-2020 14:52:20)

Results

Validation accuracy: 99.33%

Training finished: Reached final iteration

Training Time

Start time: 12-Feb-2020 14:52:20

Elapsed time: 26030 min 25 sec

Training Cycle

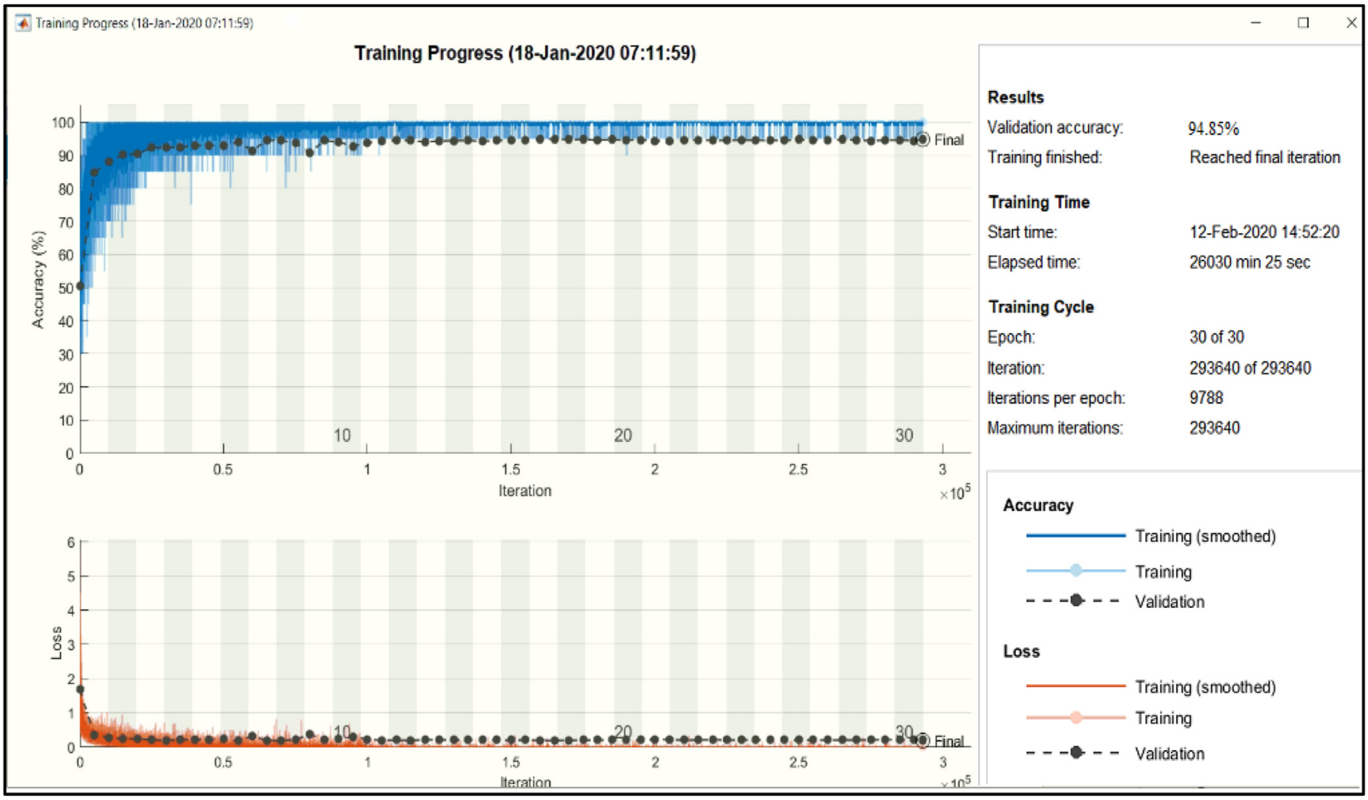
Epoch: 30 of 30

Iteration: 293640 of 293640

Iterations per epoch: 9788

Maximum iterations: 293640

Fig. 4. Progress of training the model on the high-quality part of FaceForensic++.



Training Progress (18-Jan-2020 07:11:59)

Results

Validation accuracy: 94.85%

Training finished: Reached final iteration

Training Time

Start time: 12-Feb-2020 14:52:20

Elapsed time: 26030 min 25 sec

Training Cycle

Epoch: 30 of 30

Iteration: 293640 of 293640

Iterations per epoch: 9788

Maximum iterations: 293640

Fig. 5. Progress of training the model on the low-quality part of FaceForensic++.

Table 2
Binary detection accuracy of detecting Deepfake on the test part of FaceForensic++

Methods	High Quality	Low Quality
Steganalysis features by Fridrich and Kodovsky (Fridrich and Kodovsky, 2012)	77.12	67.07
Transferable Deep-CNN by Nicolas Rahmouni et al. (Rahmouni et al., 2017)	82.16	73.25
Recasting residual-based local descriptors as CNN by Davide Conzzolino et al. (Cozzolino et al., 2017)	81.78	68.26
A deep learning using a new convolutional layer by Bayar and Stamm (Bayar and Stamm, 2016)	90.18	80.95
Mesonet by Darius Afchar et al. (Afchar et al., 2018)	95.26	89.52
XceptionNet by Francois Chollet (Chollet, 2017)	98.85	94.28
Recurrent Convolutional Strategies by Ekraam Sabir et al. (Sabir et al., 2019)	–	96.9
Our proposed model	**99.4**	**94.5**

Table 3
Binary detection accuracy of detecting Deepfake on the test part of VidTIMIT

Methods	High Quality	Low Quality
Steganalysis features by Fridrich and Kodovsky (Fridrich and Kodovsky, 2012)	83.7	75.08
Transferable Deep-CNN by Nicolas Rahmouni et al. (Rahmouni et al., 2017)	85.5	83.2
Recasting residual-based local descriptors as CNN by Davide Conzzolino et al. (Cozzolino et al., 2017)	84.15	82.43
Deep learning using a new convolutional layer by Bayar and Stamm (Bayar and Stamm, 2016)	93.5	90.78
Mesonet by Darius Afchar et al. (Afchar et al., 2018)	94.62	90.1
XceptionNet by Francois Chollet (Chollet, 2017)	98.9	95.6
Our proposed model	**99.7**	**99.2**

effectiveness of the proposed model, we have compared with the state-of-the-art forgery image detection methods performed on the same datasets, as follows: steganalysis features and SVM-based digital image classification method in (Fridrich and Kodovsky, 2012), learning-based methods used in the forensic community for manipulation image detection in (Bayar and Stamm, 2016; Cozzolino et al., 2017; Chollet, 2017), computer-generated vs. natural image detection in (Rahmouni et al., 2017), and face forgery detection in (Afchar et al., 2018).

4.3.1. *The results detecting deepfake on FaceForensic++*

Binary Deepfake detection accuracy on test part of FaceForensic++ by the proposed model and the state-of-the-art forgery image detection methods are shown in Table 2. These results show that the proposed model gives the outperform results with the binary detection accuracy of 99.4 and 94.5 on the high-quality data set and the low-quality data sets, respectively. The proposed model's result is slightly lower than the results in (Sabir et al., 2019); that is easy to understand because the proposed model has 1.3 million parameters, which is much smaller than the number of parameters in the models used in (Sabir et al., 2019).

4.3.2. *The results detecting deepfake on VidTIMIT*

The results of a binary Deepfake detection accuracy on the test part of VidTIMIT by the proposed model and the state-of-the-art forgery image detection methods are shown in Table 3. The results also show that the proposed model gives outstanding results with the binary detection accuracy of 99.7 and 99.2 on the high-quality data set and the low-quality data set, respectively. In addition, these results show that most methods give higher results than when performed detection on the FaceForensic++ dataset. It proves that the Deepfake in the VidTIMIT dataset is easier to detect than the Deepfake in the FaceForensic++ dataset.

All experiments above show that Deepfake detection using deep learning-based methods often gives more accuracy than handcrafted features. And especially the results from our proposed model give good results, outperform the state-the-art methods, and reach over 99% on all test datasets.

5. Conclusions and future directions

Nowadays, with the rapidly developing hardware industry, machine learning, and especially deep learning development. That has built many tools for automatically creating Deepfake videos that can be made for malicious purposes for public and private. But they are difficult to detect by naked eyes. Besides, so far, Deepfake videos are also challenges for automatic detection tools.

In this study, we have proposed constructing 3-D images and the 3-D model of CNN, which can learn features on spatial and temporal dimensions from the 3-D images. In experiments, the proposed model gives the outstanding detection on Deepfake FaceForensic++ and VidTIMIT datasets. Especially with high-quality videos, the proposed model's performance outperformed on both the datasets above 99%. These results are the best when comparing with the state-of-the-art methods.

In the future, we will conduct to apply the proposed model to detect the different types of facial reenactments, such as facial reenactments from Face2Face, or NeuralTextures tools.

References

Afchar, D., et al., 2018. Mesonet: a compact facial video forgery detection network. In: 2018 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE.

Bayar, B., Stamm, M.C., 2016. A deep learning approach to universal image manipulation detection using a new convolutional layer. In: Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security.

Cheng, Z., et al., 2019. Energy compaction-based image compression using convolutional autoencoder. IEEE Trans. Multimed. 22 (4), 860–873.

Chollet, F., 2017. Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Chorowski, J., et al., 2019. Unsupervised speech representation learning using wavenet autoencoders. IEEE/ACM Trans. Audio Speech Language Proc. 27 (12), 2041–2053.

Cozzolino, D., Poggi, G., Verdoliva, L., 2017. Recasting residual-based local descriptors as convolutional neural networks: an application to image forgery detection. In: Proceedings of the 5th ACM Workshop on Information Hiding and Multimedia Security.

DeepFaceLab. https://github.com/iperov/DeepFaceLab. (Accessed 1 December 2019).

Dfaker. https://github.com/dfaker/df. (Accessed 1 December 2019).

Dolhansky, B., et al., 2019. The Deepfake Detection Challenge (DFDC) Preview Dataset arXiv preprint arXiv:1910.08854.

Faceswap. Deepfakes software for all. Retrieved from. https://github.com/deepfakes/faceswap.

Faceswap. https://github.com/deepfakes/faceswap. (Accessed 1 December 2019).

Faceswap-GAN. https://github.com/shaoanlu/faceswap-GAN. (Accessed 1 December 2019).

FakeApp 2.2.0. Retrieved from. https://www/malavida.com/en/soft/fakeapp.

Fridrich, J., Kodovsky, J., 2012. Rich models for steganalysis of digital images. IEEE Trans. Inf. Forensics Secur. 7 (3), 868—882.

Güera, D., Delp, E.J., 2018. Deepfake video detection using recurrent neural networks. In: 2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). IEEE.

Ioffe, S., Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift arXiv preprint arXiv:1502.03167.

Koopman, M., Rodriguez, A.M., Geradts, Z., 2018. Detection of Deepfake video manipulation. In: Conference: IMVIP.

Korshunov, P., Marcel, S., 2018. DeepFakes: a New Threat to Face Recognition? Assessment and Detection. Idiap.

Li, Y., Lyu, S., 2018. Exposing Deepfake Videos by Detecting Face Warping Artifacts, vol. 2 arXiv preprint arXiv:1811.00656.

Li, Y., Chang, M.-C., Lyu, S., 2018. In ictu oculi: exposing ai created fake videos by detecting eye blinking. In: 2018 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE.

Matern, F., Riess, C., Stamminger, M., 2019. Exploiting visual artifacts to expose deepfakes and face manipulations. In: 2019 IEEE Winter Applications of Computer Vision Workshops (WACVW). IEEE.

Nguyen, H.H., Yamagishi, J., Echizen, I., 2019. Capsule-forensics: using capsule networks to detect forged images and videos. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE.

Punnappurath, A., Brown, M.S., 2019. Learning raw image reconstruction-aware deep image compressors. IEEE Trans. Pattern Anal. Mach. Intell. 42 (4), 1013—1019.

Rahmouni, N., et al., 2017. Distinguishing computer graphics from natural images using convolution neural networks. In: 2017 IEEE Workshop on Information Forensics and Security (WIFS). IEEE.

Rössler, A., et al., 2019. Faceforensics++: Learning to Detect Manipulated Facial Images arXiv preprint arXiv:1901.08971.

Sabir, E., et al., 2019. Recurrent convolutional strategies for face manipulation detection in videos. Interfaces 3, 1.

Srivastava, N., et al., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15 (1), 1929—1958.

Tran, D., et al., 2015. Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision.

Yang, X., Li, Y., Lyu, S., 2019. Exposing deep fakes using inconsistent head poses. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE.