Chapter 22
Localization of Facial Images Manipulation in Digital Forensics via Convolutional Neural Networks

**Ahmed A. Mawgoud, Amir Albusuny, Amr Abu-Talleb,
and Benbella S. Tawfik**

1 Introduction

The remarkable phenomenon, a troubling example of the societal threat posed by computer-generated spoofing images, is an important concern of digital forensic analysis. A spoof-video attack will make the Internet a target. Some of the tools available can be used for interpreting head and face movement in real time or for making visual images [1]. In addition, an attacker can also clone a person's voice (only a few minute are required to speak) and sync it with the visual portion for audiovisual spoofing, thanks to advances in voice synthesis and conversion [2]. Throughout the near future, such techniques will become widely accessible, so that everyone can generate profound information. In the visual domain, numerous countermeasures were introduced. Many of these were tested using only one or a few databases, including CGvsPhoto, Deepfakes and FaceForensics++ databases [3].

Cozzolino et al. [4] tackled many state-of-the-art spoofing detectors' transferability problems and developed an auto-encoding architecture to promote generalizations and easily adapt them to a new domain through easy finishing. For digital forensics, another significant issue is the location of compromised areas [5]. The shapes of the segmentation masks for the handling of facial images and video may display

A. A. Mawgoud (✉) · A. Albusuny
Information Technology Department, Faculty of Computers and Artificial Intelligence,
Cairo University, Giza, Egypt
e-mail: aabdelmawgoud@pg.cu.edu

A. Abu-Talleb
Computer Science Department, Faculty of Computers Science, University of People,
Pasadena, CA, USA
e-mail: aabdelmawgoud@pg.cu.edu.eg

B. S. Tawfik
Information System Department, Faculty of Computer Science, Suez Canal University, Ismailia,
Egypt

A. E. Hassanien et al. (eds.), *Enabling Machine Learning Applications in Data Science*,
Algorithms for Intelligent Systems, https://doi.org/10.1007/978-981-33-6129-4_22

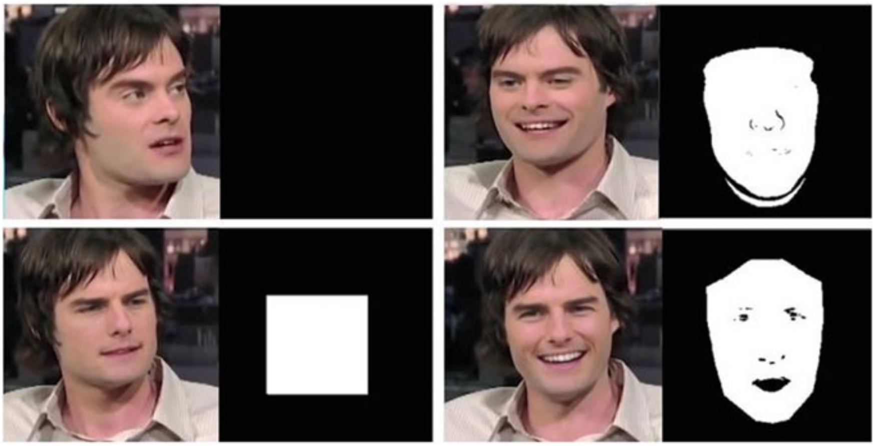

Fig. 1 Early video frame (top left), changed video frame with the Face2Face technique, using the Deepfakes technique (bottom left, rectangular mask) and using the FaceSwap method (bottom right, multiple-faceted mask)

clues as shown in Fig. 1. The three most popular forensic segmentation methods are used: elimination, copy-moving and splicing. Such methods must process full-scale images, as with other segmentation tasks of images [6].

Rahmouni et al. [1] used a sliding window to process high resolution images which Nguyen et al. [7] and Rossler et al. [8] subsequently used. In spoofing pictures produced with the Face2Face process, this sliding window effectively handles parts. However, many overlapping windows must be marked by a spoofing method which takes a lot of calculating power. We have established a multi-task method for classification and segmentation of treated facial images simultaneously. Our auto-encoder consists of an encoder and a Y-form decoder and is semi-controlled. Classification is carried out by triggering the encoded features [9]. For segmentation, the output of the decoder is used, and the other output is used to restore the input data. These tasks (classification, segmentation and reconstruction) relay information, which improves the overall efficiency of the network [10].

This paper is structured as following. Section 2 includes the related work about previous projects for facial image manipulation detections, Sect. 3 describes the proposed solution of our approach followed by Sect. 4 which is the experiment itself through using two databases and finally Sect. 5 which is the summarization of the overall paper work.

2 Related Work

2.1 Manipulated Videos Generation

The creation of a digital photo realistic actor is a fantasy for those who work in computer graphics. An initial example is the Virtual Emily project, where the image of an actress and her actions to synthesize a numerical version were recorded by sophisticated tools [11]. During the time, the perpetrators could not access this technology, so a digital representation of a victim could not be produced. It changed after Thies et al. [12] performed a face reconstruction in real time in 2016. Following research, heads with basic specifications met by any average individual could be translated. The mobile Xpression App1 was later released with the same feature.

2.2 Manipulated Images Detection Phase

For the identification of manipulated images, multiple countermeasures were introduced. A common approach is to view a video as an image series and to insert the images. The noise-based solution is one of the best developed detectors by Fridrich et al. [13]. The enhanced version with the CNN demonstrated how easily automated feature extraction can be used for detection [14]. Take advantage of high-performance pretrained models among deep learning approaches to detection, fine-tuning and transfer. This is an efficient way of enhancing the efficiency of a CNN by using the CNN as pretrained function extractor. Some detection methods include the use of a restricted convolutionary layer, use a statistical pooling layer, use a two stream network, use a lightweight CNN network and use two layers at the bottom of one CNN. Cozzolino et al. [4] have developed a benchmark for the transferability of state-of-the-art detectors for use in unseen attack detection. We also introduced an auto-encoder-like architecture, which significantly improved adaptability. Li et al. [15] proposed using a time method to detect the blindness of the eye which has not been well replicated in fake footage. Our proposed approach offers segmentation maps of controlled areas in addition to conducting classification [16]. This information could be used to assess the authenticity of images and videos particularly if the classification task does not detect spoofed inputs.

2.3 Manipulated Regions Localization

There are two common approaches to the identification of distorted areas in images: the entire image section and binary classification using a sliding window on several occasions. The segmentation technique is also used for detecting attacks on elimination, copying and splicing. For forgery segmentation, semantic segmentation approaches may also be used [17]. The boxes which represent the limits of the manipulated regions should be returned instead of returning segmentation masks as a slightly different segmentation approach. For spoofing areas, the sliding fan method is used more and creates fake images or videos from real images by a computer. In this method, binary classificators are named at any location of the sliding window, to identify images as a spoof or a good faith. The length of the sliding window can be equal to (non-overlapped) the length of the window or less than (overlapped) the length [18]. The sliding window approach is more used to detect spot areas created by a computer to produce spoof pictures or videos. In this method, binary classifiers are named at any location of the sliding window, to identify images as a spoof or a good faith. The length of the sliding window can be equal to (non-overlapped) the length of the window or less than (overlapped) the length. Our method proposed takes the first approach but with one major difference: Instead of the entire image, only the facial areas are taken into account. It overcomes the problem of measurement expenditures in large numbers inputs.

3 Proposed Method

3.1 Overview

In comparison with other single-target approaches, the likelihood of an input being spoofed and the segmentation maps of manipulated areas are shown by our proposed technique in every context of the data, as shown in Fig. 2.

A collection of frames is used to handle video inputs. For this analysis, we focused on facial images so that the face parts are preprocessed [19]. In principle, the approach proposed can accommodate different input image sizes. However, to make it simple in class, before feeding them into your auto-encoder, we resize cropped images to 256–256 pixels.

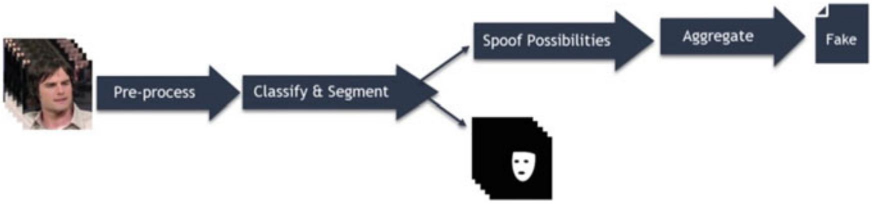

Fig. 2 Description of the formulated network

3.2 Y-shaped Auto-Encoder

The partitioning of latent characteristics and the y-figured nature of the decoder (motivated by work by Cozzolino et al. [4] enables the encoder to exchange useful information between classification, segmentation and reconstruction, enhancing overall efficiency through reduction in losses. Three specific forms of loss are activation loss D_{act}, segmentation loss D_{seg} and reconstruction loss D_{rec}. The accuracy of partitioning into the latent space is calculated by the activation of the two halves of the encoded characteristics given labels $y_i \epsilon \{0, 1\}$, activation loss:

$$L_{\text{act}} = \frac{1}{N}\sum_i \left|a_{i,1} - y_i\right| + \left|a_{i,0} - (1 - y_i)\right| \tag{1}$$

where N is the sample number, $a_{i,0}$ and $a_{i,1}$ are the activation values and the L_1 standards for half latent characteristics, $h_{i,0}$ and $h_{i,1}$ (due to K the $\{h_{i,0}\, | h_{i,1}\}$ features):

$$a_{i,c} = \frac{1}{2k}\left\|h_{i,c}\right\|_{1,} \quad c \in \{0, 1\} \tag{2}$$

This ensures that, given an input xi of class c, the corresponding half of the latent features $h_{i,\,c}$ is activated $a_{i,c} > 0$. The other half, $h_{i,1-c}$, remains quested ($a_{i,1-c} = 0$). To force the two decoders, D_{seg} and D_{rec}, to learn the right decoding schemes, we set the off-class part to zero before feeding it to the decoders ($a_{i,1-c}\colon = 0$). We are using cross-entropy loss to determine the consistency between the divisional s_i mask and the ground-truth mask m_i corresponding to the input x_i: The loss is cross-entropy loss as the segmentation loss:

$$L_{\text{seg}} = \frac{1}{N} + \sum_i \|m_i \log(s_i) + (1 + m_i)\log(1 - s_i)\|_1. \tag{3}$$

The reconstruction losses calculate the contrast between the restored image and the original image using the L_2 distance ($\dot{x}_i = D_{\text{rec}}(h_{i,0}, h_{i,1})$). The reconstruction loss is for N samples

$$L_{\text{rec}} = \frac{1}{N_{2.}}\sum_i \|x_i - \dot{x}_i\| \tag{4}$$

The weighted average of the three losses is the overall loss:

$$L_{\text{rec}} = \gamma_{\text{act}} L_{\text{act}} + \gamma_{\text{seg}} L_{\text{seg}} + \gamma_{\text{rec}} L_{\text{rec}}. \tag{5}$$

We placed the three weights on equal footing (equal to 1). Unlike Cozzolino et al. [4], the classification task and the division task are equally important, and the restoration process plays a significant role in the division mission. The results from the various settings (described below) were experimentally compared.

3.3 Implementation

As seen in Fig. 3, the *Y*-shaped auto-encoder was introduced.

It is a fully integrated CNN, using three to three convolutional windows and three to three deconstructive windows with a stride of one interspersed (for the decoder). A batch layer and a linear rectified unit (ReLU) accompany each convolutional layer.

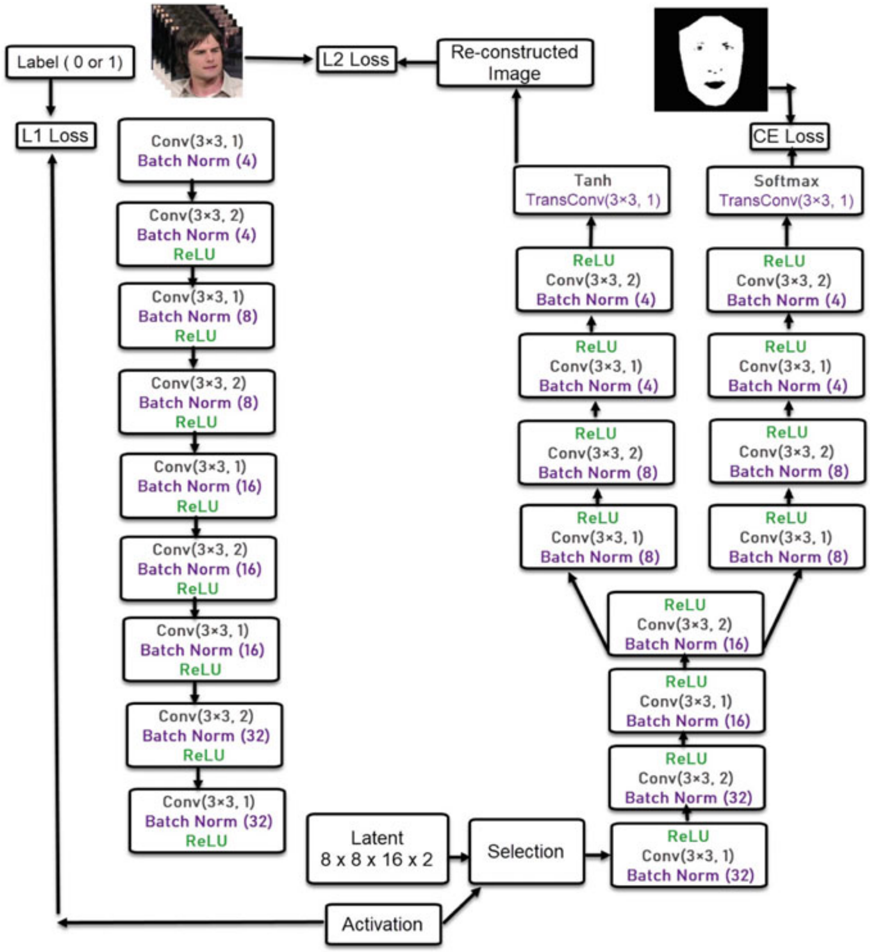
Label (0 or 1)
L1 Loss
L2 Loss
Re-constructed Image
CE Loss
Conv(3×3, 1) Batch Norm (4)
Conv(3×3, 2) Batch Norm (4) ReLU
Conv(3×3, 1) Batch Norm (8) ReLU
Conv(3×3, 2) Batch Norm (8) ReLU
Conv(3×3, 1) Batch Norm (16) ReLU
Conv(3×3, 2) Batch Norm (16) ReLU
Conv(3×3, 1) Batch Norm (16) ReLU
Conv(3×3, 2) Batch Norm (32) ReLU
Conv(3×3, 1) Batch Norm (32) ReLU
Latent 8 x 8 x 16 x 2
Selection
Activation
ReLU Conv(3×3, 1) Batch Norm (32)
ReLU Conv(3×3, 2) Batch Norm (32)
ReLU Conv(3×3, 1) Batch Norm (16)
ReLU Conv(3×3, 2) Batch Norm (16)
ReLU Conv(3×3, 1) Batch Norm (8)
ReLU Conv(3×3, 2) Batch Norm (8)
ReLU Conv(3×3, 1) Batch Norm (4)
ReLU Conv(3×3, 2) Batch Norm (4)
Tanh TransConv(3×3, 1)
ReLU Conv(3×3, 1) Batch Norm (8)
ReLU Conv(3×3, 2) Batch Norm (8)
ReLU Conv(3×3, 1) Batch Norm (4)
ReLU Conv(3×3, 2) Batch Norm (4)
Softmax TransConv(3×3, 1)

Fig. 3 Proposed *Y*-shaped auto-encoder to detect and segment manipulated facial images

Only the true half of latent characteristics ($h_{i,}$) will move through the selection block and zeros out in the other half ($h_{i,1-y}$). Thus, only the true half of latent functions is needed to decode by decoders D_{seg}; D_{rec}. The embedding dimension is 128, which has proven optimal. A softmax activation feature is used to generate segmentation maps for the segmentation branch D_{seg}. A hyperbolic tangent function (tanh) is used to form the output in the range for reconstruction branch D_{rec} $[-1; 1]$. For ease, we feed normalized images directly into the auto-encoder without turning them into residual images. Throughout further research, the advantages of the use of residual images throughout classification and segmentation tasks will be explored [20]. We trained the network after the work of Cozzolino et al. [4] with an ADAM optimizer of 0.001, a batch of 64, betas of 0.9 and 0.999 and an epsilon of 10−8. After that, we trained the network.

4 Experiments

4.1 Databases

Our proposed network has been tested using two databases:

Forensics Face and forensics Face++. The FaceForensics database contains 1004 true YouTube videos and their associated manipulated versions, divided into two sub-datasets:

- A dataset of 1004 counterfeit videos generated by the Face2Face system is repeated in the source to target; the source video (attacker) is different in each input pair for replay. Victim is different in the target pair.
- Self-reposition dataset with 1004 fake videos, generated again using the Face2Face method; the source and target videos are the same in each input pair for replication. Although this dataset does not matter from the viewpoint of the attacker, the source-to-target reenactment dataset poses a more demanding benchmark.

Every dataset was divided into 704 training videos, 150 validation videos and 150 test videos. The database also provided segmentation masks that suit the manipulated images. The H.264 codec2 is based on three compression rates, and compression light (quantization = 23) and heavy squeeze (quantization = 40) were used. The FaceForensics++ database is a modified FaceForensics database and comprises the Face2Face dataset, the FaceSwap3 dataset (graphic handling) and the Deepfakes4 dataset (deeplearning handling). There are 1,000 actual videos (1000 in each dataset) and 3000 manipulated videos. Every data package was divided into 720 training videos, 140 validation videos and 140 testing videos. For the same measurement values, the same three rates of compression based on the codec H.264 were used. We used only light-compressed videos for convenience (quantization = 23). The images from the Cozzolino et al. [4] videos have been extracted by using: 200 frames were

Table 1 Training and test datasets production

Name	Dataset source	Description	Manipulation approach	Videos number
Training	Source to target	All test usage	Face2Face	805 × 4
Test 1	Source to target	Seen attack matching	Face2Face	260 × 6
Test 2	Self-reenactment	Seen attack nonmatching	Face2Face	260 × 6
Test 3	Deepfakes	Unseen attack (deep learning)	Deepfake	240 × 4
Test 4	FaceSwap	Unseen attack (graphic based)	FaceSwap	240 × 4

used for the training of each training video, and 10 frames were used for both the evaluation and testing of each evaluation and test video. The rules for frame selection are not precise, so we have chosen the first (200 or 10) frames of each video and cut the facial areas. For the entire sample of ImageNet Large Scale Visual Recognition Challenge, we applied normalization to average (0:485; 0:456, 0:406) and standard deviation (0:229; 0:224; 0:225).

The datasets were constructed for training and research, as shown in Table 1. The Face2Face approach for generating manipulated videos has been used for test 1 and test 2 datasets. The pictures in test 2 were more difficult to identify than in test 1, because the source and objective videos for re-enacting were the same, and the reenactment pictures were of higher quality. Therefore, the match and the conditions of a clear attack are called Test 1 and Test 2. The Deepfake attack method was used in Test 3, while the FaceSwap attack method was used in Test 4, presented in the FaceForensics++ database. Both of these attack strategies have not been used to build the training set and were thus known as invisible attacks.

Table 2 Auto-encoder configuration

No.	Approach	Depth	Seg. Weight	Rec. Weight	Rec. Loss
1	*FT res*	Narrower	0.3	0.3	L_1
2	*FT*	Narrower	0.3	0.3	L_1
3	*Deeper FT*	Deeper	0.3	0.3	L_1
4	*Proposed old*	Deeper	0.3	0.3	L_1
5	*No recon*	Deeper	2	1	L_2
6	*Proposed*	Deeper	2	1	L_2

4.2 Training Y-Shaped Auto-Encoder

We built the configuration as shown in Table 2 to determine the contribution of each part in the Y-shaped auto-encoder. The methods of FT Res and FT are re-implementation with or without residual images by Cozzolino et al. [4]. We can also be interpreted without a segmentation branch as the Y-shaped auto-encoder. Lower FT is a lower, the same depth variant of FT as the proposed process. The old approach is the method proposed by using Cozzolino et al. [4] weighting settings. Researchers demonstrated that they can detect and identify image manipulations, outperforming human observers significantly. They use recent advances in deep learning, particularly with convolutional neural networks that can learn extremely powerful image features. Training a neural network allows them to tackle the detection issue. For this purpose, they collect a wide dataset of model-based methods for manipulation.

We have equipped the shallowest networks with 100 epochs, and the lower networks with 50 epochs because we need to converge longer than deeper networks. In each process, all the tests mentioned in this section were performed using the training stage of maximum accuracy for the classification task and an adequate segmentation loss (if available).

4.3 Dealing with Identified Attacks

Table 3 (Test 1) and Table 4 (Test 2), respectively, display the results for match and incompatibility conditions for seen attacks. The deeper networks (the last four) were much better defined than the lower networks.

The provided results in the proposed method in Cozzolino et al. [4] showed that that the detection rate in the manipulated images was not effective enough when it is compared with the previous contributions. The new methods used by the new settings for this segmentation task were higher than the old system, which used the old weighting settings, for the segmentation task.

When dealing with the mismatch condition for seen attacks, the efficiency of all methods was slightly degraded. The FT Res and the new methods proposed were best

Table 3 Test 1—image results

Method	Classification		Segmentation Acc (%)
	Acc (%)	EER (%)	
FT_Res	64.6	54.2	_
FT	62.31	52.88	_
Deeper_FT	64.45	48.11	_
Proposed_Old	**67.93**	47.31	95.34
No_Recon	65.97	46.97	**95.97**
Proposed_New	65.08	**45.05**	95.78

Table 4 Test 2—image results

Method	Classification		Segmentation Acc (%)
	Acc (%)	EER (%)	
FT_Res	64.6	54.2	_
FT	62.31	52.88	_
Deeper_FT	64.45	48.11	_
Proposed_Old	**67.93**	47.31	95.34
No_Recon	65.97	46.97	**95.97**
Proposed_New	65.08	**45.05**	95.78

adapted, as shown in their lower degradation scores. It illustrates the significance of using the residual (FT Res) images and the reconstruction (new weighting proposed approach for the *Y*-shaped auto-encoder). The reconstruction branch also helped to get the highest score for the segmentation task in the proposed new system.

4.4 Dealing with Un-identified Attacks

4.4.1 Evaluation Through Pretrained Model

All six approaches had slightly lower precision and higher EERs for invisible attacks, as shown in Table 5 (Test 3) and Table 6 (Test 4). In test 3, the shallower approaches, in particular FT Res, had greater adaptability. The more profound approaches, the tests of almost random grouping had a higher probability of overfitting [21]. Test 4 indicated in its best EERs that the decision thresholds were moved, although all the methods suffered from almost random classification accuracies. The findings of the segmentation were a fascinating finding. Although degraded, it still showed high segmentation accuracies, particularly in Test 4 in which FaceSwap was using a computer graphic method to copy the facial area from source to target. This knowledge for segmentation may also provide an important indicator in order to determine

Table 5 Test 3—image results

Method	Classification		Segmentation Acc (%)
	Acc (%)	EER (%)	
FT_Res	64.6	54.2	_
FT	62.31	52.88	_
Deeper_FT	64.45	48.11	_
Proposed_Old	**67.93**	47.31	95.34
No_Recon	65.97	46.97	**95.97**
Proposed_New	65.08	**45.05**	95.78

Table 6 Test 4 before fine-tuning—image results

Method	Classification		Segmentation Acc (%)
	Acc (%)	EER (%)	
FT_Res	64.6	54.2	_
FT	62.31	52.88	_
Deeper_FT	64.45	48.11	_
Proposed_Old	**67.93**	47.31	95.34
No_Recon	65.97	46.97	**95.97**
Proposed_New	65.08	**45.05**	95.78

the validity of the queried images while dealing with unexpected attacks [22] (Table 6).

4.4.2 Fine-Tuning Through Limited Data

For the finalizing of all methods, we used a FaceForensics + -FaceSwap validation package (a small set which is usually used to select hyperparameters during training which vary from the test set). To make sure the frequency data was small, for every video, we used only ten frames. The dataset was split into two parts: 100 videos from each training class and 40 for each assessment class. We trained them in 50 years and picked the best models based on their evaluation results.

Table 7 displays the results after the test 4 is done. Their classification and segmentation accuracies increased in relation to the small amount of data, respectively, by approximately 20 and 7%. The one exception was the old approach introduced—it did not enhance the segmentation accuracy.

The FT Res method was much better adapted than the FT system, which supports the argument of Cozzolino et al. [4] as demonstrated by the results of Table 7, the latest approach proposed had the most potential transferability against unknown attacks.

Table 7 Test 4 after fine-tuning—image results

Method	Classification		Segmentation Acc (%)
FT_Res	90.05 (37.65)	28.68 (**36.64**)	_
FT	81.91 (29.71)	36.67 (27.34)	_
Deeper_FT	93.11 (39.72)	28.4 (20.78)	_
Proposed_Old	89.68 (32.86)	31.81 (26.61)	95.41(1.27)
No_Recon	83.84 (39.18)	27.14 (29.14)	83.71(8.85)
Proposed_New	**94.82** (**30.75**)	**26.18** (29.08)	95.12(9.45)

5 Conclusion

The proposed neuronal network with a Y-shaped auto-encoder demonstrated its effectiveness, which is widely used by classifiers, for both classification and segmentation tasks without using a slack window. Sharing data among classification, segmentation and reconstruction tasks improved the overall performance of the network, in particular for malfunctions seen. However, by using only a few fine-tuning samples, the auto-encoder is easy to change for unseen attacks. Future research should specifically investigate how the residual images will influence the output of the auto-encoder, process high resolution images with no re-dimension and enhance its ability to deal with invisible attacks and enlarge it to the domain of the audiovisual.

References

1. Rahmouni N, Nozick V, Yamagishi J, Echizen I (2017) Distinguishing computer graphics from natural images using convolution neural networks. In: 2017 IEEE workshop on information forensics and security (WIFS). IEEE, pp 1–6
2. Mawgoud AA, Taha MH, Khalifa NE, Loey M (2020) Cyber security risks in MENA region: threats, challenges and countermeasures. In: International conference on advanced intelligent systems and informatics. Springer, Cham, pp 912–921
3. Gong D (2020) Deepfake forensics, an ai-synthesized detection with deep convolutional generative adversarial networks. Int J Adv Trends Comput Sci Eng 9:2861–2870. https://doi.org/10.30534/ijatcse/2020/58932020
4. Rossler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2019) Faceforensics++: learning to detect manipulated facial images. In: Proceedings of the IEEE international conference on computer vision, pp 1–11
5. Kleinmann A, Wool A (2014) Accurate modeling of the siemens S7 SCADA protocol for intrusion detection and digital forensics. J Digit Foren Secur Law. https://doi.org/10.15394/jdfsl.2014.1169
6. Mawgoud A, Ali IA (2020) Statistical insights and fraud techniques for telecommunications sector in Egypt. In: international conference on innovative trends in communication and computer engineering (ITCE). IEEE, pp 143–150
7. Nguyen HH, Fang F, Yamagishi J, Echizen I (2019) Multi-task learning for detecting and segmenting manipulated facial images and videos. arXiv preprint arXiv:1906.06876
8. Rössler A, Cozzolino D, Verdoliva L, Riess C, Thies J, Nießner M (2018) Faceforensics: a large-scale video dataset for forgery detection in human faces. arXiv preprint arXiv:1803.09179
9. Mohan A, Meenakshi Sundaram V (2020) V3O2: hybrid deep learning model for hyperspectral image classification using vanilla-3D and octave-2D convolution. J Real-Time Image Process. https://doi.org/10.1007/s11554-020-00966-z
10. Zarghili A, Belghini N, Zahi A, Ezghari S (2017) Fuzzy similarity-based classification method for gender recognition using 3D facial images. Int J Biometrics 9:253. https://doi.org/10.1504/ijbm.2017.10009328
11. Saini K, Kaur S (2016) Forensic examination of computer-manipulated documents using image processing techniques. Egypt J Foren Sci 6:317–322. https://doi.org/10.1016/j.ejfs.2015.03.001
12. Thies J, Zollhofer M, Stamminger M, Theobalt C, Nießner M (2016) Face2face: real-time face capture and reenactment of rgb videos. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 2387–2395

13. Mawgoud AA, Taha MHN, Khalifa NEM (2020) Security threats of social internet of things in the higher education environment. In: Toward Social Internet of Things (SIoT): enabling technologies, architectures and applications. Springer, Cham, pp 151–171
14. Fridrich J, Kodovsky J (2012) Rich models for steganalysis of digital images. IEEE Trans Inf Foren Secur 7(3):868–882
15. Bayar B, Stamm MC (2018) Constrained convolutional neural networks: a new approach towards general purpose image manipulation detection. IEEE Trans Inf Foren Secur 13(11):2691–2706
16. El Karadawy AI, Mawgoud AA, Rady HM (2020) An empirical analysis on load balancing and service broker techniques using cloud analyst simulator. In: 2020 international conference on innovative trends in communication and computer engineering (ITCE), Aswan, Egypt (2020), pp 27–32
17. Chu CC, Aggarwal JK (1993) The integration of image segmentation maps using region and edge information. IEEE Trans Pattern Anal Mach Intell 15(12):1241–1252
18. Wei Y, Feng J, Liang X, Cheng MM, Zhao Y, Yan S (2017) Object region mining with adversarial erasing: a simple classification to semantic segmentation approach. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1568–1576
19. Marshall, S.W., Xerox Corp, 2004. Method for sliding window image processing of associative operators. U.S. Patent 6,714,694.
20. Terzopoulos D, Waters K (1990) Analysis of facial images using physical and anatomical models. In: Proceedings 3rd international conference on computer vision, pp 727–728. IEEE Computer Society
21. Mawgoud AA (2020) A survey on ad-hoc cloud computing challenges. In: 2020 international conference on innovative trends in communication and computer engineering (ITCE), pp 14–19. IEEE; Heseltine T, Pears N, Austin J (2002), July. Evaluation of image preprocessing techniques for Eigen face-based face recognition. In: 2nd international conference on image and graphics, vol 4875. International Society for Optics and Photonics, pp 677–685.
22. Korshunova I, Shi W, Dambre J, Theis L (2017) Fast face-swap using convolutional neural networks. In: Proceedings of the IEEE international conference on computer vision, pp 3677–3685