

Detection of Deep Fake in Face Images Using Deep Learning

Hanady Sabah Abdul kareem^{1*} and Mohammed Sahib Mahdi Altaei¹

¹Computer Science Department, College of Science, Al-Nahrain University, Baghdad, Iraq

*Corresponding Author: Hanady Sabah Abdul kareem

DOI: <https://doi.org/10.31185/wjcm.92>

Received: September 2022; Accepted: November 2022; Available online: December 2022

ABSTRACT: Fake images are one of the most widespread phenomena that have a significant influence on our social life, particularly in the world of politics and celeb. Nowadays, generating fake images has become very easy due to the powerful yet simple applications in mobile devices that navigate in the social media world and with the emergence of the Generative Adversarial Network (GAN) that produces images which are indistinguishable to the human eye. Which makes fake images and fake videos easy to perform, difficult to detect, and fast to spread. As a result, image processing and artificial intelligence play an important role in solving such issues. Thus, detecting fake images is a critical problem that must be controlled and to prevent these numerous harmful effects. This research proposed utilizing the most popular algorithm in deep learning is (Convolution Neural Network) to detect the fake images.

The first steps includes a preprocessing which start with converting images from RGB to YCbCr color space, after that entering the Gamma correction. finally extract edge detection by entering the Canny filter on them. After that, utilizing two different method of detection by applying (Convolution Neural Network with Principal Component Analysis) and (Convolution Neural Network without Principal Component Analysis) as a classifiers.

The results reveal that the use of CNN with PCA in this research results in acceptable accuracy. In contrast, using CNN only gave the highest level of accuracy in detecting manipulated images.

Keywords: Deep learning, Deep fake, Generative Adversarial Network, Con- volution Neural Network, Principal Component Analysis



1. INTRODUCTION

Due to the technological development and the wide spread of programs for deep fakes on digital images with the advent of Generative Adversarial Network (GAN). Image and video editing are becoming easier and considered a cybercrime due to the fake information and images can be spread fast and widely in Internet through social media. Thus, deepfake can be considered as the ability to automatically create, modify, or swap a person's face in videos and images using algorithms depending on Deep Learning technology, that is one of the phenomena that is expanding quickly. It is feasible to produce top-notch results by developing new multimedia materials that are difficult for the human eye to distinguish between real and fake. This term "Deepfake" refers to all multimedia items that have been synthetically modified or produced using generative machine learning models [1]. DeepFake is concerning because it combines a high level of authenticity, quick evolution and pervasiveness [2]. On Reddit, DeepFake has been widely utilized as of November 2017. due to the United States widespread manufacturing of pornographic videos that has gained a solid online reputation and attracted interest of the people from all walks of life. In January DeepFake was officially used in an application in 2018. As a result DeepFake proliferation was accelerated. The object of face swapping has also grown in popularity from celebrities and politicians to students, friends and coworkers As a result, people's panic were naturally prompted. The main IT companies have also

begun to collaborate action with the academic community to avoid additional detrimental impact on the fierce discussion about whether and how to control Deepfake technology [3].

With the continuous enhancement of the computing power of computer, the ongoing reduction of hardware price and the high integration of deep learning tools like tensorflow [4] and keras [5], technicians with specific professional backgrounds can produce high-quality faked images and videos consistent with the real distribution of data through the use of convolution automatic encoder [6] and Generative adversarial networks (GAN) [7]. Deepfake image as display in Figure 1



FIGURE 1. Several illustrations of DeepFake: (a) Obama, made by BuzzFeed in cooperation with Monkeypaw Studios; (b) Mark Zuckerberg, made by artists Bill Posters and Daniel Howe in association with advertising company canny ; (c) Matteo Renzi,, made by “Striscia la Notizia” [8].

2. RELATED WORK

1. Y. Wang et al in 2021 presented two algorithms for detection of fakeface images. The first approach is the Local Binary Pattern (LBP)-Net using global texture features used to detect fakefaces. The second method ensemble model constructed from five models including LBP-Net, Gram-Net, ResNet and two models utilizing InceptionResnetV1 pre-trained on Casia-Webfaceare and vggface2. Results of detecting fakeface images by several image augmentation such as downsample (66.32%), brightness (81.09%), Solarize (75.04%) ,Contrast (85.42%) and color (91.06%) when using “140K Real and Fake Faces” [9].
2. M. Taeb et al in 2022 compared the most popular state-of-the-art face-detection classifiers such as CNN, VGG19, and DenseNet-121 using an enhanced actual and fakeface dataset. Data augmentation is a technique for improving performance with conserving computing resources. When compared to other studied models, early findings show that VGG19 has the best performance and accuracy of 95%. When using “140K Real and Fake Faces” [10].
3. S. Tariq and et al. in 2018 proposed an ensemble of three different convolution neural network with different layers as a classifier after performing pre-processing represented by face cropping and noise filtering methods. Finally a fully automated end-to-end fakeface detection pipeline developed to be focused on image content with only RGB channel information in order to recognize GAN-created face images with 94% AUROC score (Area Under the Receiver Operating Characteristic curve) and recognize human-created fake face images with 74.9 % AUROC score. Where, applied on CelebA and PGGAN dataset images. [11].
4. X. Chang et al in 2020 improved VGG network termed NA-VGG, it is used to detect deepfake face images, which was based on image noise and augmentation of image. The SRM filter layer is utilized to highlight the noise features of the image and after that the image noise is acquired as the network’s input. Second, the image noise map is augmented to make the face features appear weaker. Finally, the augmented noise of images are fed into the network, which is trained and used to determine whether or not the image is fake. The obtained accuracy is 85.7 % when using the Celeb-DF dataset [12].

3. THE WORKFLOW OF THE MODEL

This project has been performed in five steps. The broad discussion of these stages is described here. The first stage is choosing the suitable real and fake images dataset from kaggle.com and preprocessing the dataset. following that, PCA applying for selecting images features after splitting the dataset by using cross-validation (hold-out) (80:20). The next stage is using (CNN) classifiers to classify the dataset. Finally evaluate performance of model using various metrics such as (accuracy, precision, recall, and F1-score). as showed in Figure 2.

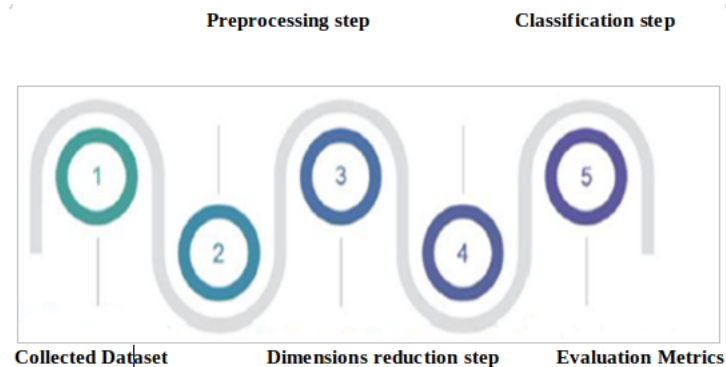


FIGURE 2. Work design step of fake images detection.

The Proposed Deep Fake Detection

3.1 DATASET

Deepfake detection systems often use binary classifiers to group information into fake and real classes. This strategy needs a great quantity of high-quality authentic and manipulated data to train the models of classification. The dataset was taken from kaggle website. This dataset contains of all 70k real faces from of the Flickr dataset gathered by Nvidia, in addition 70k fake faces picked from the Bojan's 1 Million fake faces (produced using StyleGAN).

In this dataset, combined both dataset, scaled all of the images to 256px, and divided the data into three sets: train, validation and test set. also some CSV files available for convenience [13]. In this study, just two features (images and labels) are utilized to detect fake image classifiers. Label one represents fake images, whereas label zero represents true images.

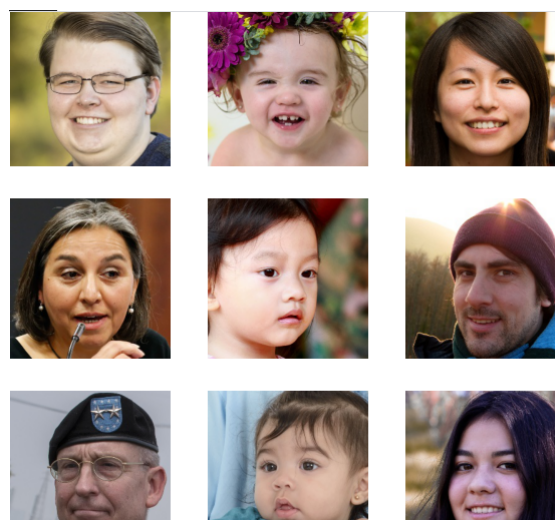


FIGURE 3. Samples of the dataset's real images

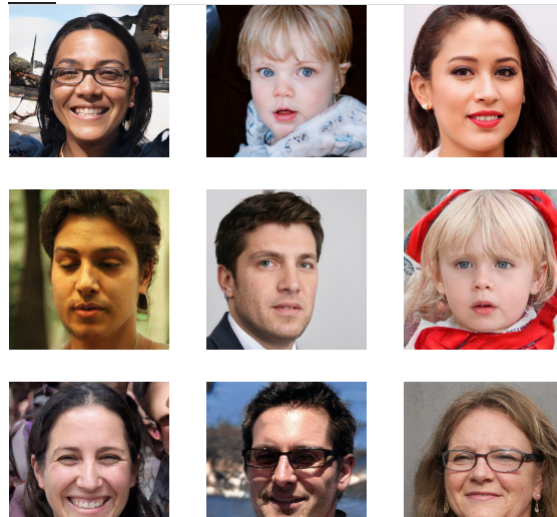


FIGURE 4. Samples of the dataset's fake images

3.2 PRE-PROCESSING STEPS

In this section the results of the pre-processing of the images were reviewed, where six images were taken randomly from the dataset that was used in the proposed system. The first three represent the real images and the second three represent the fake ones, where the figure shows the results before and after the pre-processing. The first column represents the original images with RGB color space, while the second column represents the images conversion from RGB into YCbCr color space, the third column represents the images after entering the Gamma correction and finally the fourth column represents the images after entering the Canny filter on them. As displayed in Figure 5.

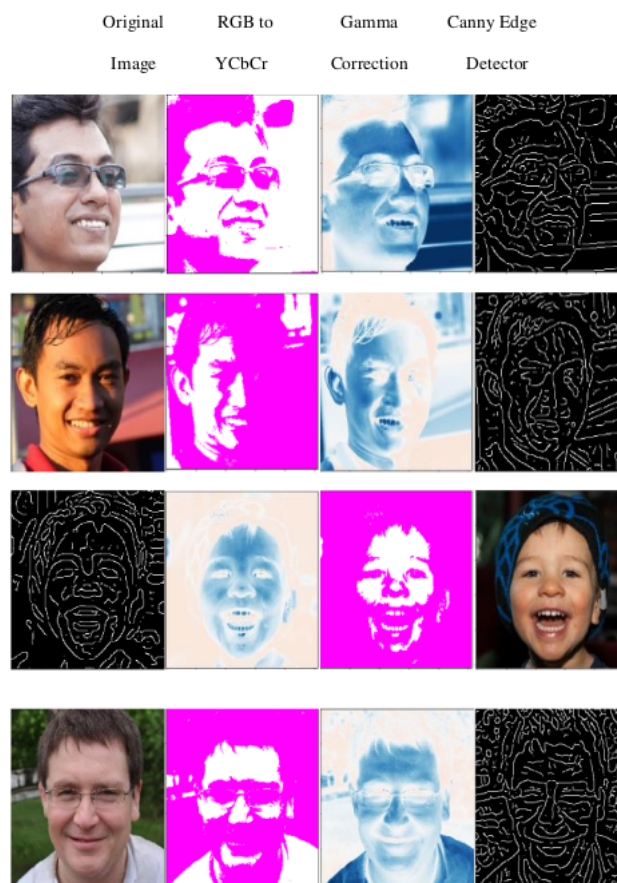




FIGURE 5. Resulted of a preprocessing sample image from 140k Real and Fake Faces .

3.3 PRINCIPAL COMPONENT ANALYSIS (PCA)

Principal Component Analysis (PCA) is one of the most widely statistical approaches used for feature selection. It has several uses in picture compression, text classification, and face recognition [14]. This is a frequently used method for reducing the dimensionality of a feature collection via a linear transformation. The main goal of PCA is to reduce the original variables to subset of variables by calculating the highest relationship of the original variables [15]. Although the resultant dataset is reduced, but the original data set's features are still retained and removed the redundancy of information [16] and [17]. The number of features in the new dataset may be equal to or lower than that in the original dataset. The principal components are computed by using the covariance matrix. The ability discriminative of the classifiers can be improved by using PCA .

The following steps summarize the process [18].

1. Let the training set of images $\{X_1, X_2, X_3 \dots X_N\}$. Calculation of mean value of an image with equation:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (1)$$

Information:

x_i = data variable

n = numbers of data

Calculation of the mean value to reduce the dimension to be calculated in the next process.

2. Compute the covariance matrix to depict the scatter degree of all feature vectors connected to the average vector. The covariance matrix C is defined as follows:

$$C = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(x_i - \bar{x})^T \quad (2)$$

3. Calculate the Eigenvectors and eigenvalues from covariance matrix

$$CV = IV \quad (3)$$

Where V is the set of eigenvectors correlated with its eigenvalue .

4. Sorts the eigenvector and eigenvalues from high to low based on the order of eigenvalues.

The main component k is the eigenvector corresponding to the largest eigenvalue.

The main component k of the vector x is observed using the equation:

$$w_i = v_i^T (x_i - \bar{x}) \quad (4)$$

Algorithm (1): Principal Component Analysis (PCA)
Input: Images Output: components
Begin Step 1: Form 2 dimensional matrix of independent variables X. Rows represent data items and columns represent features. The number of columns is the number of dimensions. Step 2: Subtract the mean of each column Step 3: Standardize your data Divide each value in a column by the column's standard deviation, which is the centered and standardized matrix Z. Step 4: Get Covariance of Z Estimate the covariance matrix of Z, $CovMat = Z^T Z$ Step 5: Calculate Eigen Vectors E and Eigen Values λ Step 6: Sort the Eigen Vectors Take the eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_p$ and sort them from largest to smallest. In doing so, sort the eigenvectors in E accordingly. (For example, if λ_3 is the largest eigenvalue, then take the third column of E and place it in the first column position.) Call this sorted matrix of eigenvectors E^* . The columns of P^* are the same as the columns of P in a different order. Note that these eigenvectors are independent of one another. Step 7: Calculate the new features Calculate $Z^* = ZE^*$. Step 8: Drop unimportant features from the new set Keep the first feature column to form (PCA). END

3.4 CLASSIFIER

Convolutional Neural Networks is a type of Artificial Neural Net-work (ANN), also known as Conv-Nets or CNN, That is one of the most effective deep designs for classifying image data and many other applications, including audio recognition, object detection, medical image analysis and natural language processing [19]. The deep architecture of the network produces hierarchical feature extraction, wherein the trained filters of the first layer can be seen as a set of color dots or edges, of the second layer as some forms, the filter of the next layer may learn part of objects and the final layers filters may be able to recognize the objects. [20].

In the proposed method the design is carried out in two stages. Which starts first by training the model on the dataset consisting of real images and fake images. As for the second stage which is the testing stage, it is easy to distinguish the test image whether it is fake or real. One of the most common algorithms used in classification is a convolutional neural network (CNN) architecture by created a model from scratch with using six blocks in the CNN architecture. In each block utilize Conv2D with kernel size=5, Max Pooling, batch normalization=64, model = Sequential, dropout, activation = Relu, padding = same, epsilon=0.001, epochs = 100 One iteration on each training data set is represented as an epoch, verbose = 1, shuffle=True and to optimize the network using Adam optimizer = 0.001 and a learning rate 0.0001. As discussed in Table 1.

Once obtaining the features (components) the features are fed to Convolution Neural Network (CNN) layer of the deep

Table 1. CNN architectures for deep fake detection of face image. Each row represents a block in the architecture. (Note: Conv=Conv2D, BN=BatchNormalization, DO=Dropout, MP=MaxPooling, F=Flatten, GAP=GlobalAveragePooling, D=Dense & S=Sigmoid)

Convolution Neural Network
Conv-ReLU-Padding-MP-BN
Conv-ReLU-Padding-MP-BN-DO
Conv-ReLU-Padding-MP-BN-DO
Conv-ReLU-Padding-MP-BN-DO
Conv-ReLU-Padding-MP-BN-DO
Conv-ReLU-Padding-MP-BN-DO
GAP-F-D-ReLU-DO
D- ReLU-DO-D-S

learning model, which further selects the useful features by using their filters. The selected features are supplied to the max-pooling layer to choose the features that have the highest importance value throughout the computation. Gradients will be calculated, and the network's weights will automatically adjust.. We report our detection accuracy in result section.

3.5 EVALUATION METRICS

In order to evaluate the accuracy of algorithm's classification in detecting fake images, a number of evaluation metrics were used. The most commonly used measuring metric (Confusion Matrix) for detecting fake images has been applied in this part. It is possible to specify the metrics that the confusion matrix will use by defining this as a classification task. has as below [21].

$$\text{Accuracy} = \frac{TP+FP}{TP+TN+FP+FN} \quad [22]$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad [23]$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad [24]$$

$$\text{F1 Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad [25]$$

where TP represents (True Positive) and TN represent (True Negative). Moreover, FP is False positive, and FN is False-negative. As discussed in Table 2.

Table 2. Parameters of evaluation metrics

Parameters	Description
TP True Positive	The number of correctly classified records
TN True Negative	The number of the correctly rejected of records that have been classified
FP False Positive	The number of incorrectly classified records
FN False Negative	The number of the incorrectly rejected of records that have been classified

These measures are often used in a sequence of machine learning algorithms to evaluate the performance of a classifier from several estimations. Specifically, the accuracy measure, which quantifies the similarity between predicted and actual fake images. Precision refers to the measuring of the fraction of the found fake images, that has been classified as fake, addressing the critical issue of the fake images classification. However, because to the dataset of fake images is typically skewed, good precision may be achieved by making a lower number of optimistic predictions, hence recall is used to assess sensitivity, or the fraction of annotated fake articles predicted as fake. It should be highlighted that higher values refers to better Recall, Precision, and Accuracy [26]. Finally F1 score": Another term for it is the F Score or F Measure. The F1 score is a great balance of P and R [25].

4. RESULTS AND DISCUSSION

According on the classification results showed that the accuracy of the model with pre-processing is CNN only without PCA and CNN with PCA classifier is 63.86% and 74.26%, respectively. Table 3, Table 4, Table 5, Table 6, Table 7 and Table 8 displays the resulted of confusion matrix with TP, FP, TN and FN values. Our experimental findings without preprocessing stages show that CNN only without PCA achieves 93.16% and CNN with PCA achieves 90.76%. Finally, an additional experiment was conducted by increasing the number of samples entering the CNN network (Training and Testing) which gave the highest accuracy results in image classification, so that the classification accuracy reached 98.04%.

$$\text{Accuracy} = 63.86$$

(a) CNN without PCA with preprocessing

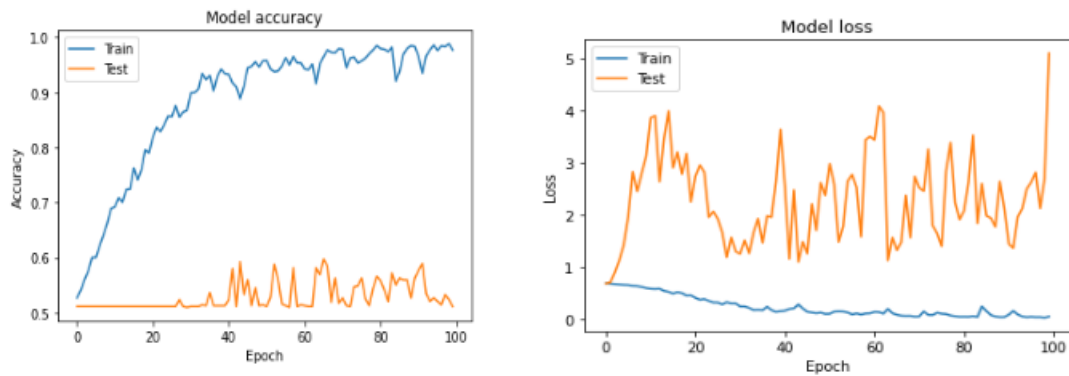


FIGURE 6. The curves of the validation accuracy and loss during the training stage for the proposed CNN without PCA with preprocessing.

Table 3. Confusion matrix of the detection of deep fake on face image using CNN only without PCA with preprocessing

	Positive	Negative
Positive	1064	436
Negative	648	852

Precision=70.93

Recall=62.14

F1 Score=66.24

(b) CNN with PCA with preprocessing

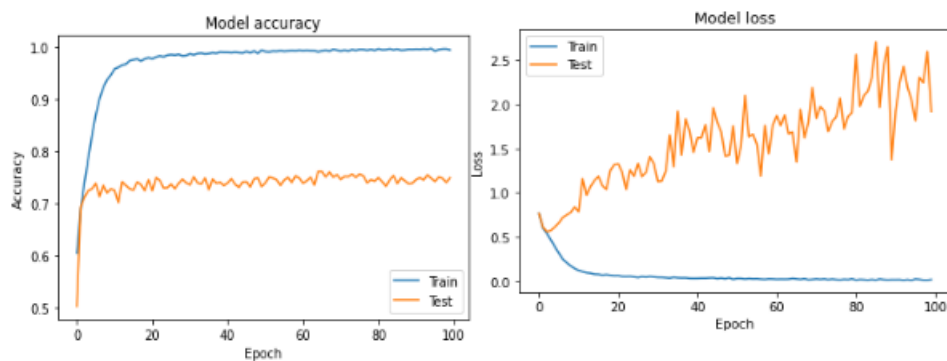


FIGURE 7. The curves of the validation accuracy and loss during the training stage for the proposed CNN with PCA with preprocessing.

Table 4. Confusion matrix of the detection of deep fake on face image using CNN with PCA with preprocessing

	Positive	Negative
Positive	1144	356
Negative	416	1084

Accuracy =74.26

Precision=76.26

Recall=73.33

F1 Score=74.76

Accuracy =93.16

Precision=94.26

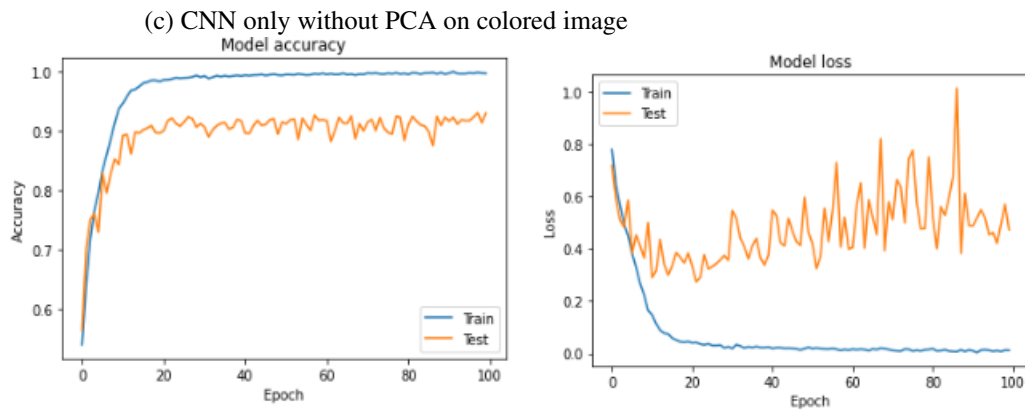


FIGURE 8. The curves of the validation accuracy and loss during the training stage for the proposed CNN only.

Table 5. Confusion matrix of the detection of deep fake on face image using CNN only without PCA on colored image

	Positive	Negative
Positive	1414	86
Negative	119	1381

Recall=92.23
F1 Score=93.23

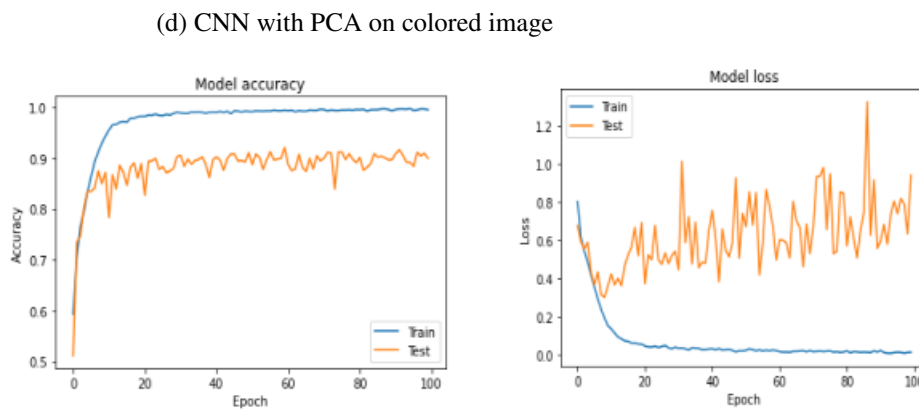


FIGURE 9. The curves of the validation accuracy and loss during the training stage for the proposed CNN with PCA.

Table 6. Confusion matrix of the detection of deep fake on face image using CNN with PCA on colored image

	Positive	Negative
Positive	1373	127
Negative	150	1350

Accuracy =90.76
Precision=91.53
Recall=90.15
F1 Score=90.83

Table 7. Confusion matrix of the detection of deep fake on face image using CNN only without PCA on colored image

	Positive	Negative
Positive	7833	167
Negative	146	7854

(e) CNN only without PCA on colored image

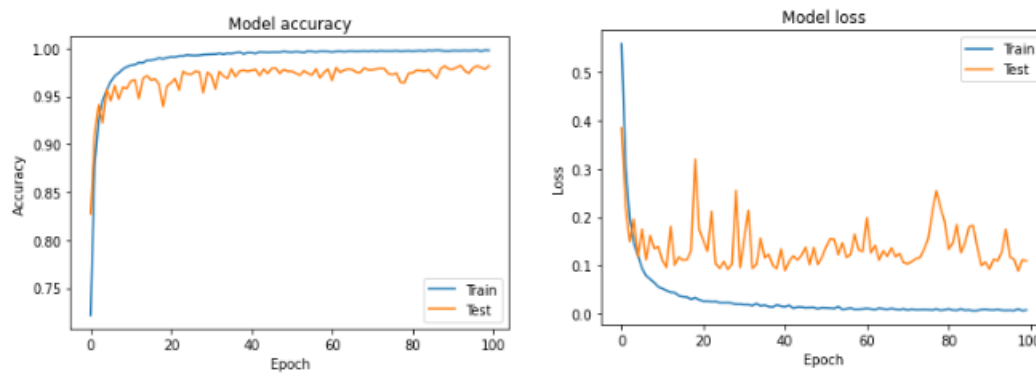


FIGURE 10. The curves of the validation accuracy and loss during the training stage for the proposed CNN.

Accuracy =98.04
 Precision=97.91
 Recall=98.17
 F1 Score=98.03

(f) CNN with PCA on gray image

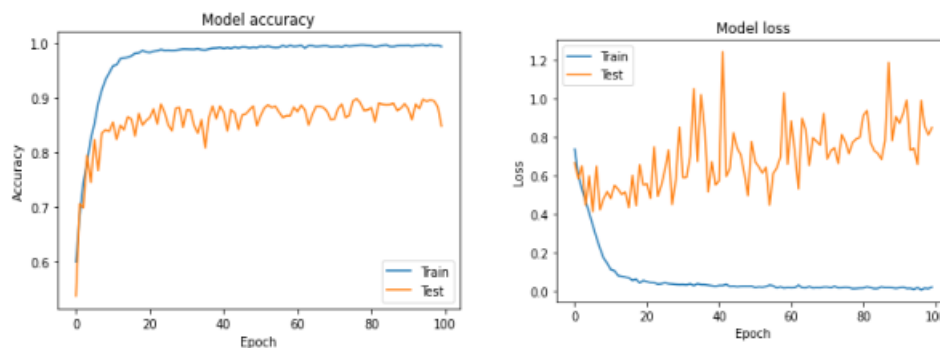


FIGURE 11. proposed CNN with PCA on gray image.

Table 8. Confusion matrix of the detection of deep fake on face image using CNN with PCA on gray image

	Positive	Negative
Positive	1304	196
Negative	209	1291

Accuracy =86.5
 Precision=86.93
 Recall=86.18
 F1 Score=86.55

In this section, the results were reviewed for implementing the proposed system and the results were as shown in the explanation below.

The research aims to reveal the manipulation that may be present in the images of faces, and therefore it was necessary for us to do a test stating that preserving the image entered into the system for the purpose of testing it and detecting the presence of manipulation in it first. As a result of the initial processing of the image, it may lead to the burial of some traces of manipulation in the image.

So we tested the image by using the technique involved the use of deep learning technology. In this case, we used successive steps for a simple preprocessing (determining the size of the image).

We clarified the image and reveal its edges before entering the tamper detection system.

We found that any preprocessing of the image, led to undesirable results compared to the results obtained without preprocessing.

Where the network of detection of manipulation in faces using the CNN classifier with the preprocessing and the use of canny Detection to detect edges was about 63.86

An additional experiment was used by adding the PCA method after the stage of preprocessing, which produced detection rates of 74.26% for the CNN classifier.

This indicates that the use of the PCA as a feature selector has improved the results of detection by a small percentage, but it is not high, because the work of the PCA depends on converting the data of the raw image entered into the detection system into components containing more information that spreads downward from the first component to the last component. The use of a limited set of component. Since the PCA was used in the form of a feature selector, it relied on the first components in the classification process by CNN, leaving the last components that contain information but are very weak and represent a burden on the classification process.

High classification rates and proof that any pre-processing on the image leads to changing or erasing the traces of manipulation encouraged the use of classification techniques approved by CNN, which is CNN with the PCA and without pre-processing, the detection rate of manipulation reached 90.76%, while detection rates using CNN only reached to 93.43. A new attempt was also used by inserting the CNN classifier with PCA on a gray scale image and the results were 86.5%.

Finally, CNN classifiers were used to detect forgery and applied to the images directly without using PCA and without any pre-processing, but with more data to see if the network is affected by the number of data that is trained on, and the classification results were 98.04% for the CNN classifier, which is the highest accuracy result in image classification It reached by the proposed system

This indicates that the use of PCA in the proposed system affected the accuracy of the results, and that the process of inserting direct images into the CNN classifier has greatly benefited in the process of detecting fraud, as CNN needs a lot of information for the purpose of training and learning from it, regardless of the percentage of information contained in it. Pictures. CNN also made use of the images as more raw data. From the case that the PCA application has, that is, it made more use of the images than the components

5. CONCLUSION

The remarkable development of artificial intelligence and with the efficiency of the GAN in generating fake images that are closer to reality, it was necessary to find an efficient way to detect fake images. the final result of our research was to confront the phenomenon of Deepfake offered a detection model for the fake images by utilizing the most popular algorithm in deep learning is (Convolution Neural Network) to detect the fake images. The preprocessing steps start with converting images from RGB to YCbCr color space, after that entering the Gamma correction. finally extract edge detection by entering the Canny filter on them. After that, utilizing two different method of detection by applying (CNN with PCA) and (CNN without PCA) as a classifiers.

The achieved result is better than the listed related work, so utilizing this method enhances the accuracy of classification . From the above results, we conclude the following:

1. The CNN only is better than a CNN with PCA in the classification accuracy of fake images dataset.
2. CNN is more suitable for large datasets because the network efficiency increases with more data, that is, it gives better results as the training data increases
3. Preprocessing steps when using our dataset give worse results. These steps had a huge impact on decreasing classification accuracy.
4. The type of data used has a significant impact on the categorization accuracy of this work.

FUNDING

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

- [1] L. Guarnera, O. Giudice, and S. Battiato, "Deepfake detection by analyzing convolutional traces," *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 666–667, 2020.
- [2] L. S. Wang, 2019.
- [3] X. Chang, J. Wu, T. Yang, and G. Feng, 2020.
- [4] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, and M. Devin, "{TensorFlow}: a system for {Large-Scale} machine learning," *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pp. 265–283, 2016.
- [5] F. Chollet, 2015.
- [6] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt, "Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction," *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 1274–1283, 2017.
- [7] G. Antipov, M. Baccouche, and J.-L. Dugelay, "Face aging with conditional generative adversarial networks," *2017 IEEE international conference on image processing (ICIP)*, pp. 2089–2093, 2017.
- [8] L. Guarnera, O. Giudice, C. Nastasi, and S. Battiato, "Preliminary forensics analysis of deepfake images," *2020 AEIT international annual conference (AEIT)*, pp. 1–6, 2020.
- [9] Y. Wang, V. Zarghami, and S. Cui, "Fake Face Detection using Local Binary Pattern and Ensemble Modeling," *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 3917–3921, 2021.
- [10] M. Taeb and H. Chi, "Comparison of Deepfake Detection Techniques through Deep Learning," *Journal of Cybersecurity and Privacy*, vol. 2, no. 1, pp. 2022–2022.
- [11] S. Tariq, S. Lee, H. Kim, Y. Shin, and S. S. Woo, "Detecting both machine and human created fake face images in the wild," *Proceedings of the 2nd international workshop on multimedia privacy and security*, pp. 81–87, 2018.
- [12] X. Chang, J. Wu, T. Yang, and G. Feng, 2020.
- [13]
- [14] A. I. Taloba, A. Dalia, Eisa, S. I. Safaa, and Ismail, 2018.
- [15] Karamizadeh, S. M. Sasan, A. A. Abdullah, M. Manaf, A. Zamani, and Hooman, "An overview of principal component analysis," *Journal of Signal and Information Processing*, vol. 4, 2020.
- [16] M. Ahmad, A. M. Khan, J. A. Brown, S. Protasov, and A. M. Khattak, "Gait fingerprinting-based user identification on smartphones," *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 3060–3067, 2016.
- [17] M. Ahmad, D. I. Ul, Q. Haq, M. Mushtaq, and Sohaib, "A new statistical approach for band clustering and band selection using K-means clustering," *Int. J. Eng. Technol*, vol. 3, no. 6, 2011.
- [18] V. Arjun, R. R. Mane, V. Manza, and Karbhari, "Human face recognition using superior principal component analysis (SPCA)," *International Journal of Computer Theory and Engineering*, vol. 2, no. 5, 2010.
- [19] B. Traore, B. Boubacar, F. Kamsu-Foguem, and Tangara, "Deep convolution neural network for image recognition," *Ecological Informatics*, vol. 48, 2018.
- [20] N. Aloysius and M. Geetha, "A review on deep convolutional neural networks," *2017 international conference on communication and signal processing (ICCSP)*, pp. 588–592, 2017.
- [21] J. Ali, R. Khan, N. Ahmad, and I. Maqsood, "Random forests and decision trees," *International Journal of Computer Science Issues (IJCSI)*, vol. 9, no. 5, 2012.
- [22] P. Shahane and D. Gore, "Detection of Fake Profiles on Twitter using Random Forest & Deep Convolutional Neural Network," *International Journal of Management*, 2019.
- [23] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of economic perspectives*, vol. 31, no. 2, 2017.
- [24] S. Gilda, "Notice of violation of IEEE publication principles: Evaluating machine learning algorithms for fake news detection," *2017 IEEE 15th student conference on research and development (SCOREd)*, pp. 110–115, 2017.
- [25] J. Han, M. Kamber, and J. Pei, "Data mining concepts and techniques third edition," *The Morgan Kaufmann Series in Data Management Systems*, vol. 5, pp. 83–124, 2011.
- [26] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD explorations newsletter*, vol. 19, no. 1, 2017.