Al-Salam Journal for Engineering and Technology
Journal Homepage: http://journal.alsalam.edu.iq/index.php/ajest
e-ISSN: 2790-4822p-ISSN: 2958-0862

Detection of Deep Fake in Face Images Based Machine Learning

Hanady Sabah Abdul kareem[1]*, Mohammed Sahib Mahdi Altaei [1]

[1]Department of Computer Science, College of Science, Al-Nahrain University, Baghdad, IRAQ

*Corresponding Author: Hanady Sabah Abdul kareem

DOI: https://doi.org/10.55145/ajest.2023.02.02.001
Received August 2022; Accepted December 2022; Available online February2023

ABSTRACT: Fake face images is a recent critical issue of artificial intelligent due to it has directly impacts on the social lives, and may be made to imply threats against privacy, fraud, and other issues. Currently, creating fake images has become relatively simple due to the powerful yet user-friendly mobile applications that navigate in the social media world and with the invention of the Generative Adversarial Network (GAN) that provides a good quality images that might be difficult for humans to differentiate with their eyes and makes image and video manipulation simple to do, quickly spread and hard to detect, Therefor, image processing and artificial intelligence are crucial in solving such problems. That is why scientists must create technologies or algorithms to control and to avoid these various negative impacts by different detection approaches can be applied. The proposed approach is more robust than current methods when propose a model based on support vector machine as a classifiers to detect fake human faces created by machines. The first stages includes a preprocessing that start with changing images from RGB to YCbCr and then applying the gamma correction. finalize the results show that the extracted edges using Canny filter were useful for detecting fakes in face images. After that, applying two distinct methods of detection by utilizing "Support Vector Machine" with "Principal Component Analysis" and "Support Vector Machine" without "Principal Component Analysis" as a classifiers. The findings show that the highest accuracy gained is 96.8% when using the SVM with PCA while the accuracy obtained is 72.2% when using the SVM alone.

Keywords: Principal Component Analysis, Support Vector Machine, Machine Learning, Generative Adversarial Network

1. INTRODUCTION

In general, the great development has occurred in artificial intelligence programs and machine learning. In particular, the increase in cybercrime which reached 67% in recent years. One of the most critical issues that national security systems have had to cope with is security breaches[1]. It has become possible to generate and upload petabytes of data over the internet every second. A deep fake detection approach is used to effectively detect fake content to know the supplied content is real or fake [2]. Thus, avoiding deepfakes is a difficult process [3]. So, deepfakes can be defined as a synthetic media and a deep learning approaches to create false photos and videos by overlaid one person's face on top of another person's face in an already existing image or video [4]. Images of fake faces are often produced using Generative Adversarial Networks (GANs), which are high-quality, sophisticated images. As a result, recognition of this type of image is extremely difficult and is not directly possible with the naked eye [5]. The example of Deepfake images is shown in Fig. 1.

**Corresponding author: hanadysabah763@gmail.com*

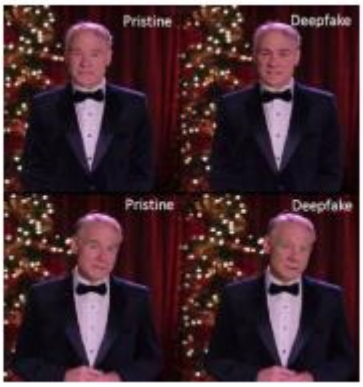


FIGURE 1. - deepfake face images of Ian McKellen Tommy and Lee Jones [6]

In the traditional technique to image forgery detection, there are two categories:

- External feature
- Internal feature

First, external unique signals will be embedded in the original images to identify fraud (e.g. digital watermarking). Then, evaluate the extracted watermark against the original watermark to see if the received image is a fake or not. The second technique will scan the source photos for basic and invariant characteristics. The fake image must be capable of being recognized. Through the statistical characteristics of the intrinsic feature that was obtained from the received picture. In general, In the first category the original extrinsic signal is required of any tampering technique to determine whether it is fake or not. Obtaining an original extrinsic signal (i.e. watermark) is difficult for each incoming image. The second technique, on the other hand, is only looking for characteristic inherent in the incoming image. such as an uncommon statistical property, to determine if it is a fake or not [7]. There are numerous methods for finding inherent features of images in order to identify altered photos [7] and [8].

fraud detection approach in [8] finds sensor pattern distortion as the intrinsic feature. uses double compressing cues as an intrinsic feature for JPEG formatted images [9]. Despite significant advancements, have been made to identify image fraud, it actually still a challenging task becaus e the majority of modern strategies rely on deep learning, which requires massive amounts of labeled data. This study will discuss the detection of deepfake on face images with Machine Learning. Although deepfakes also can take the form of audio and video. our major goal is to detect deepfakes using image data.

The contribution of this research is trying to enhance the accuracy findings of the detection of deepfake face images by using PCA as a feature selection and fed the PCA components to the SVM as a classifier. And then compare the results in detecting fake images between using SVM alone and using it with PCA. In the following, section 2 introduces the related work. Section 3 illustrates the model's operational process. Section 4 describes the proposed system. Section 5 discusses the experimental results. Finally section 6 shows the conclusions.

2. RELATEDWORK

- L. Guarnera et al in 2020 suggested extracting a set of local attributes by using an Expectation Maximization (EM) algorithm to describe the convolutional traces left in pictures. The produced feature vector was used as input for several "naïve" classifiers were trains to differentiate between real pictures and images created by the realistic architectures on the CELEBA dataset. The deep fake detection scores were about:92.67% "CELEBA" Vs. "ATTGAN" by KNN, 88.40% "CELEBA" Vs. "GDWCT" by KNN, 93.17% "CELEBA" Vs. "STARGAN" by SVM, 99.65% "CELEBA" Vs. ."STYLEGAN" by KNN and 99.81% "CELEBA" Vs. "STYLEGAN2" by SVM[10].
- R. Rafique et al in 2021 used Error Level Analysis "ELA"-based "Deep Learning" techniques to distinguish between false and real photos. The photos first from the dataset are normalized them to 255*255 pixels, next utilizing "ELA" to assess compression ratio of the image, after that forward image to two "CNN" models i.e. Alex Net and Shuffle Net for image classification. Finally passing the feature vector to "KNN" and "SVM" classifiers. The presented work achieved the perfect accuracy of 88.2% of "Shuffle-Net" via "KNN" and "Alex-Net" vector had the accuracy of 86.8% via "KNN", while the accuracy of "Shuffle Net" via "SVM" was 87.9% and "Alex Net" vector had the accuracy of 86.1% via "SVM" [11].

- Y. Wang et al in 2021 presented two algorithms for detection of fake face images. The first approach is the Local Binary Pattern (LBP)-Net using global texture features used to detect fake faces. The second method ensemble model constructed from five models including LBP-Net, Gram-Net, Res-Net and two models utilizing Inception, ResnetV1 pre-trained on Casia - Webfaceare and vggface2. Results of detecting fake face 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" [12].

- M. Taeb et al in 2022 compared the popular state-of-the-art face-detection classification techniques like as "CNN", "VGG19", and "DenseNet-121" utilizing an augmented actual and fake face detection 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 greatest accuracy and efficiency of 95%, while the DenseNet achieved 94%. Finally the accuracy of custom CNN was 89%. When using "140K Real and Fake Faces"[13].

The result of the proposed method in current work was compared with deepfake detection technique of other previous works as illustrated in Table 1.

Table 1. – Comparison of proposed method with previous works

References	Year	Method	Dataset	Accuracy %
[10]	2020	features extracted through the EM algorithm. The produced feature vector was used as input for several "naïve" (KNN, SVM and LDA)	CELEBA against (GDWCT, STARGAN, ATTGAN, STYLEGAN, STYLEGAN2) dataset	92.67%, 88.40%, 93.17%, 99.65%, 99.81%
[11]	2021	Error Level Analysis to assess compression, then forward image to Alex Net and Shuffle Net. Finally passing the feature vector to "KNN" and "SVM" classifiers.	Real and Fake Face detection	88.2% Shuffle-Net with KNN 86.8% Alex-Net with KNN 87.9% Shuffle Net with SVM 86.1% Alex Net with SVM
[12]	2021	Local Binary Pattern, ensemble model from five models including LBP-Net, Gram-Net, ResNet and two models utilizing Inception, ResnetV1	140K Real and Fake Faces	Downsample66.32% brightness 81.09% Solarize 75.04% Contrast 85.42% color 91.06%
[13]	2022	VGG19, DenseNet-121 Custom CNN	140K Real and Fake Faces	95% 94% 89%
Proposed method	2023	PCA as a feature selection and pass the PCA components to the SVM as a classifier	140K Real and Fake Faces	96.8%

3. THE MODEL'S OPERATIONAL PROCESS

This project required five steps for the completion. Here is a description of the broad discussion of these stages. Selecting the appropriate real and fake images dataset from (kaggel.com) is the initial step and preprocessing the dataset. After dividing the dataset using cross-validation (hold-out) (80:20), the PCA is applying to choose image features. The dataset will then be classified using (SVM) classifiers in the following step. Finally, as shown in Fig. 2. Evaluate the model's performance using several metrics, including "accuracy", "recall", "precision" and "F1-score".

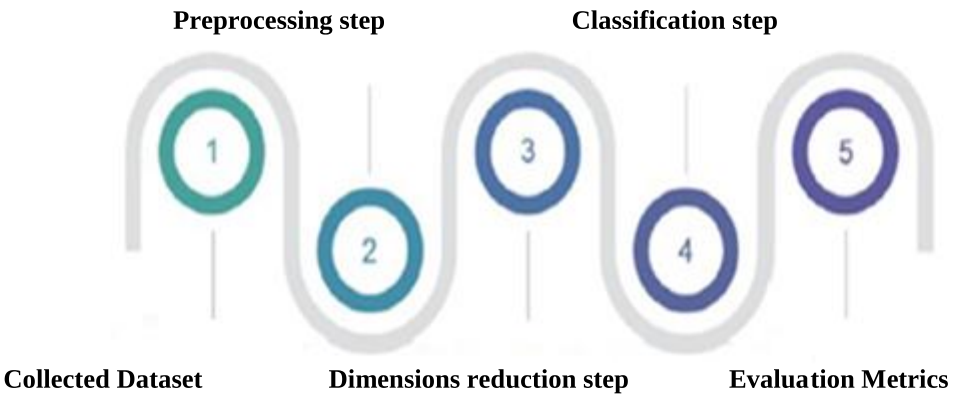


FIGURE 2. - steps in the work process of detecting fake images

4. THE PROPOSED DEEP FAKE DETECTION

4.1 DATASET

Binary classifiers are widely used by deepfake detection systems to categorize information into (fake and real) classes. To train the classification models in this strategy, a significant quantity of real and fake high-quality data is required. Dataset obtained from the Kaggle website. This dataset comprises of all (70k real) faces out from "Flickr" dataset collected by "Nvidia". Also, (70k fake) faces were chosen from "Bojan's 1 Million" fake faces (A product made with StyleGAN). In this dataset, both datasets were combined, all of the images were scaled to 256 pixels see Figures. 3 and 4, the data was split into three sets: train, validation, and test. For convenience, there are also some CSV files available [14]. Only two features: images and labels—are used in this study to identify fake image classifiers. Label zero denotes real images, while label one denotes fake images.

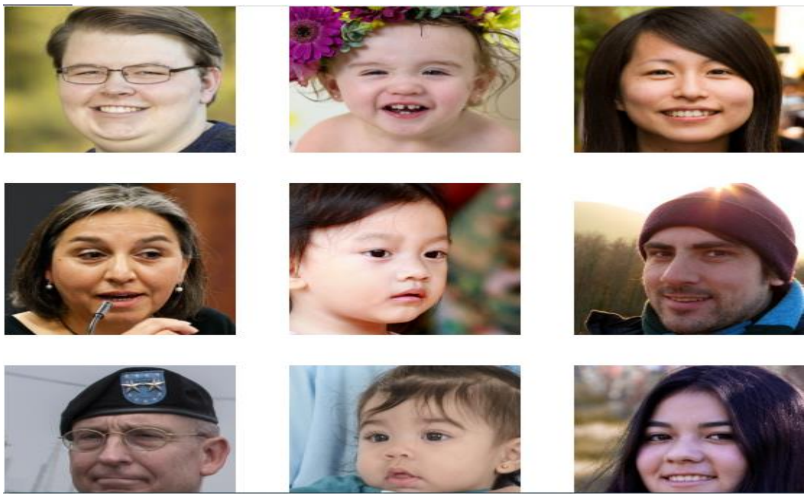
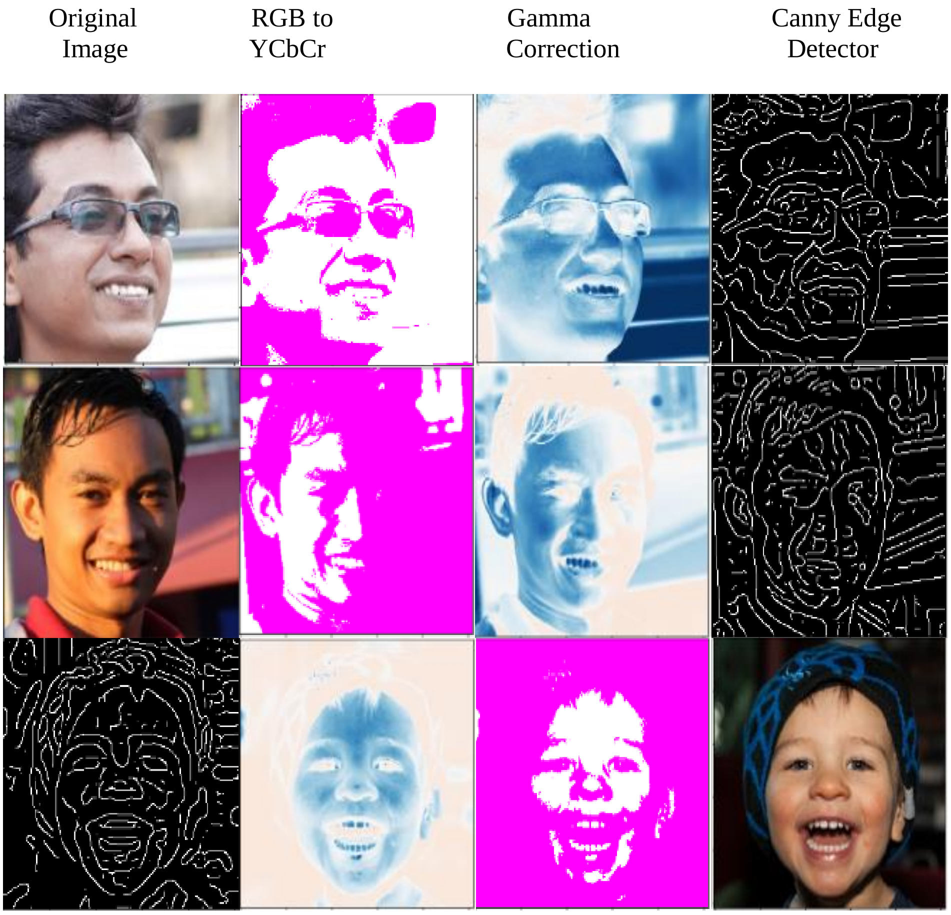


FIGURE 3. - excerpts of the real images dataset

FIGURE 4. - excerpts of the fake images dataset

4.2 PRE-PROCESSING STEPS

In this part present the performance of the pre-processing on the images. Six randomly selected images from the dataset used to develop the suggested system. The images displays the outcomes before and after the pre-processing. The first three images are real images but the following three are fake. The first column displays the picture as they were originally created in "RGB" color space, while the second column displays how the pictures were converted from RGB to YCbCr color space, the third column displays how the pictures after applying gamma correction. The images after entering the Canny filter are shown in the fourth column. As displayed in Fig. 5.



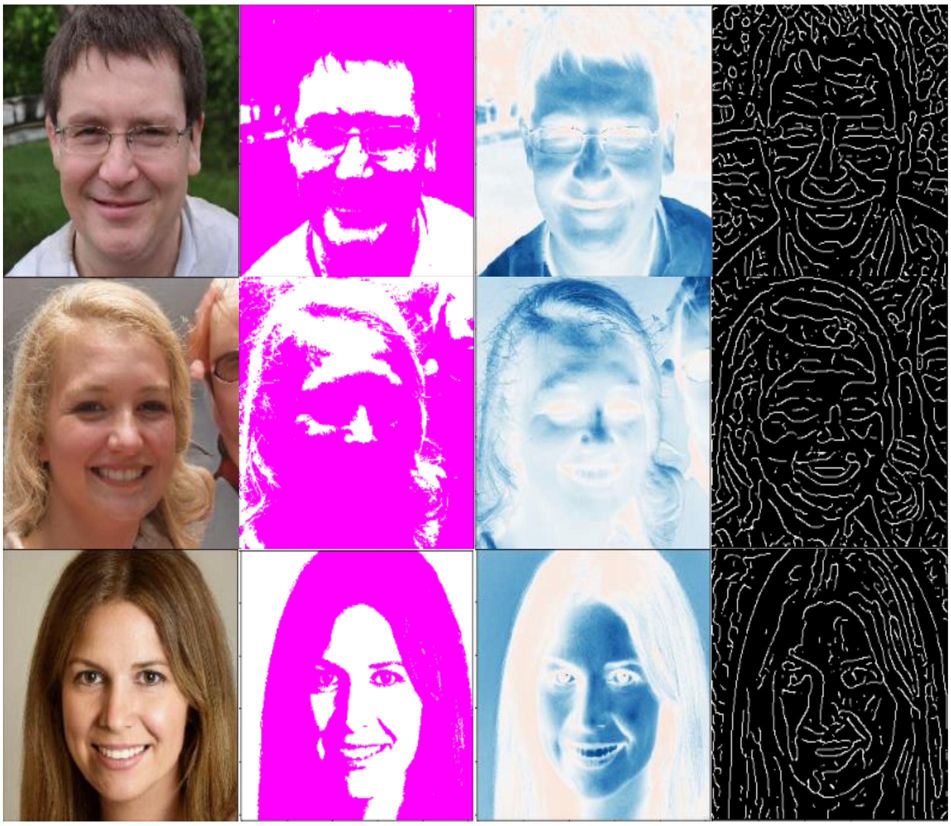


FIGURE 5. - results of a preprocessing sample image from "140k Real and Fake Faces"

4.3 "PRINCIPAL COMPONENT ANALYSIS" (PCA)

"Principal Component Analysis" (PCA): an example of the statistical techniques utilized for features selection. It has multiple uses in text classification, picture compression, and face recognition [15]. This approach for linearly reducing the dimensionality of a feature dataset is widely utilized. , PCA's primary objective is to reduce the input variables to a number of variables by determining the original variables' strongest correlation [16]. Although the final dataset is reducer, the features of the original data set are kept and redundant information is eliminated [17] and [18]. The new dataset's feature count might be the same as or less than that of the original dataset. The covariance matrix is used to compute the principal components. Using PCA can enhance the classifiers' discriminative abilities.
The following stages summarize the procedure [19].

- Let the training set of images {X1, X2, X3… XN}. Equation for calculating the mean value of an image:

  $$\overline{x} = \frac{1}{n}\sum_{i=1}^{n} xi \qquad \text{….(1)}$$

  Information:
  xi = data variable
  n = numbers of data
  Calculation of the "Mean" value to decrease the dimension that has to be calculated in the following step.
- To depict the scatter degree of each feature vector associated to the average vector, compute the "Covariance Matrix." The definition of the "Covariance Matrix" C is as follows:

  $$c = \frac{1}{n}\sum_{i=1}^{n}(xi - \overline{x})(xi - \overline{x})^T \qquad \text{…. (2)}$$
- Calculate the "Eigenvectors" and "Eigenvalues" from "Covariance Matrix".

  $$CV = lV \qquad \text{…. (3)}$$

  Where V is the set of "Eigenvectors" correlated with its "Eigenvalue" .
- 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:

  $$wi = vi^T(xi - \overline{x}) \qquad \text{…. (4)}$$

4.4 CLASSIFIER ALGORITHMS

An essential method used in machine learning is the classifier algorithm, which organizes or categorizes data into at least one or a set of classes [20]. Support Vector Machines a "supervised" machine learning algorithm that is one of the most effective and widely utilized for classifying data[21-22].

SVM's basically objective is to create the maximum marginal hyperplanes in multidimensional space as possible to separate between different classes. The error can be kept to minimum by creating the hyperplane iteratively.

Important SVM concepts include:

- Support Vectors: Which are the data points closest to the hyperplan and have the greatest impact on the hyperplane position.
- Hyperplane: It is the optimum decision boundary which is utilized to separate or split the classes in n-dimensional space. The process of creating hyperplane in such a manner which it has the maximum margin.
- SVM Kernels: SVM method is implement with Kernel to convert an incoming data space into the desired shape. Through adding more dimensions, the kernel transforms non-separable issues into separable issues [23].

The following is a list of four basic kernels [24]:

- Linear : X (yi,yj) = $yi^T yj$ ….(5)
- Polynomial: X (yi,yj) = (yi^T yj + r $)^d$ ….(6)
- Radial basis function : X {yi,yj) = exp (-γ‖‖yi-yj ‖ ….(7)
- Sigmoid : X (yi,yj) =tanh (γ $yi^T yj$+ r $)^d$ ….(8)

Where, γ,r, and d are kernel parameters.

SVM, which was utilized to train the feature file, is used to perform the classification task. The use of Grid Search function to tune the SVM hyperparameters, resulting in a combination of SVM hyperparameters auto-tuning between a set of parameters C, kernel, degree and gamma values. It is found that the best results were obtained when the following parameters were set: C =2.5, Kernel = polynomial, Degree =3, Gamma=Auto.

Then, using the SVM fit approach, the features (an array of pixel values) of the input training images are used to build a model that will ultimately make classification decisions.

```
Algorithm (2): Support Vector Machine (SVM)
Input: features (Pixels or PCA components)
Output: Decision of SVM (Real or Fake)
Begin
    Step (1): Read the dataset.
    Step (2): Preprocessing on the dataset using ( Image color transform + Gamma correction + Canny edge detection )
    Step (3): Convert the images into features using PCA
    Step (4): Divide data into training and testing sets
    Step (5): Assign cost parameter  C ← 2.5
                        Kernel K ← Polynomial
                        Degree D← Three
    Step (6): Classify training data set using current kernel K
    Step (7): The best value of C and D are obtained
    Step (8): select the testing dataset
    Step (9): Implement the test and prediction
    Step (10): Calculate testing errors using ( Confusion Matrix )
END
```

4.5 EVALUATION METRICS

Several evaluation metrics were employed to assess the classification accuracy of the system in identifying fake images. In this section, the most used statistic (Confusion Matrix) for detecting fake images has been utilized. By classifying this as a classification task, the confusion matrix will employ as seen below. Where TP denotes for (True Positive) and TN for (True Negative). Additionally, FP denotes for (False Positive) and FN for (False Negative). According to Table 2. Parameters of evaluation metrics.

Table 2. - Parameters of evaluation metrics

Parameters	Description
TP (True Positive)	the total number of successfully classified images that were fake.
TN (True Negative)	the total number of successfully classified images that were real.
FP (False Positive)	The amount of genuine images that were mistakenly labeled as fake
FN (False Negative)	The amount of fake images that were mistakenly labeled as real

These metrics are frequently used in a series of machine learning algorithms to assess how well a classifier performs given a variety of estimates. The accuracy metric, which expresses how closely expected and actual fake pictures resemble one another. Precision is the measurement of the percentage of the discover fake images, that has been labeled as fake, addressing the crucial challenge of the categorization of fake images. Since the collection of fake images is generally skewed, making it possible to attain acceptable precision by making fewer pessimistic predictions. Recall is particularly utilized to assess sensitivity, or the proportion of annotated fake images that were correctly identified as fake. In particular, higher numbers indicate improved recall, precision, and accuracy [25]. 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[24].

5. RESULTS AND DISCUSSION

Based on the classification findings showed that the accuracy of the classifier with pre-processing is SVM only without PCA and SVM with PCA classifier is 63.86% and 74.26%, respectively. (Table3, Table4, Table5 and Table6) present the outcomes of confusion matrix with TP, FP, TN and FN values. The findings without preprocessing stages demonstrate that SVM only without PCA achieves 72.2% and SVM with PCA achieves 96.8%.

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

	Positive	Negative
Positive	407	93
Negative	213	287

Accuracy =69.39

Precision=81.4

Recall=65.64

F1 Score=72.67

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

	Positive	**Negative**
Positive	363	137
Negative	113	387

Accuracy =75.0

Precision=72.6

Recall=76.26

F1 Score=74.38

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

	Positive	**Negative**
Positive	418	82
Negative	196	304

Accuracy =72.2

Precision=83.6

Recall=68.07

F1 Score=75.03

Table 6. - Confusion matrix of the detection of deep fake on face image using SVM only with PCA

	Positive	**Negative**
Positive	475	0
Negative	32	493

Accuracy =96.8

Precision=100

Recall=93.68

F1 Score=96.73

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 the method involved the use of machine 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 model of detection of manipulation in faces using the SVM classifier with the preprocessing and the use of canny Detection to detect edges was about 69.39%

An additional experiment was used by adding the PCA method after the stage of preprocessing, which produced detection rates of 75.0% for the SVM 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 SVM leaving the last components that contain information but they are very weak and represent a burden on the classification process.

The high rates of classification and the proof that any manipulation of the image leads to a change or to the erasure of the effects of manipulation encouraged the use of the approved classification techniques, which is the SVM without the preprocessing, where the detection rate of manipulation of the SVM classifier with the use of PCA was up to 96.8%.

While, the SVM classifier was used to detect forgery and applied to the images directly without using the PCA and without any preprocessing. The classification results were 72.2%.

This indicates that the use of PCA in the machine learning process, i.e. the SVM, was very useful, as only the useful attributes of the SVM classifier were used, and the useless attributes were abandoned by PCA when the unimportant components were abandoned, which indicates that these components were They intersect with each other and have directions that do not correspond to the orientation of the SVM. this result achieves the goal of the research.

6. CONCLUSION

Due to the astonishing advancement of machine learning and with the effectiveness of the GAN in producing more realistic fake images. It was required to face the phenomena of deepfake by presenting a model for detecting fake images using (SVM with PCA) . Which is Our research's ultimate goal. The SVM method is a powerful machine learning method for identifying false images. The preprocessing phase started by converting the images from RGB to YCbCr color format before commencing the Gamma correction stage of preprocessing. Finally, add the Canny filter to them to extract edge detection.

Next, two distinct detection methods SVM with PCA and SVM alone without PCA were used. The results show that the best achieved accuracy is equal to 96.8% when combining SVM with PCA, while the achieved accuracy is equal to 72.2% when using SVM alone. The average training run time is about 21 minutes and 45 seconds for the proposed system.

Therefore, PCA with SVM outperformed in detecting faces that were manipulated during classifier training processes, and therefore it will be adopted as a good descriptor in the following work.

Thus we draw the following conclusions from the findings above:

- SVM with PCA is better than SVM only in the accuracy of classification the fake images dataset.
- The pre process on dataset utilizing in this paper provides less desirable outcomes. The decrease in classification accuracy was significantly impacted by these steps of preprocessing.
- Increasing the accuracy of this work's categorization is significantly impacted by the type of data used.
- PCA features selection method showed less features impurity, the close features have been removed to reduce the overlap and spacing between features, the number of components to be 100, in which there is a 90% of the changes in the data are transfer to the resulted components.
- Frequent training showed that the false results of the classification were due to some spectral factors related to the nature of imagery system.
- Not employing data augmentation simplifies the process without impacting the model's predictive capabilities.

ACKNOWLEDGEMENT

Acknowledgements and Reference heading should be left justified, bold, with the first letter capitalized but have no numbers. Text below continues as normal.

CONFLICTS OF INTEREST

The authors declare no conflict of interest

REFERENCES

[1] Ferreira, Sara, Mário Antunes, and Manuel E. Correia. "Exposing Manipulated Photos and Videos in Digital Forensics Analysis." *Journal of Imaging* 7, no. 7, 2021.

[2] Sharma, Jatin, Sahil Sharma, Vijay Kumar, Hany S. Hussein, and Hammam Alshazly. "Deepfakes Classification of Faces Using Convolutional Neural Networks." *Traitement du Signal* 39, no. 3 2022

[3] Roets, Arne. "'Fake news': Incorrect, but hard to correct. The role of cognitive ability on the impact of false information on social impressions." *Intelligence* 65, 2017.

[4] Tolosana, Ruben, Ruben Vera-Rodriguez, Julian Fierrez, Aythami Morales, and Javier Ortega-Garcia. "Deepfakes and beyond: A survey of face manipulation and fake detection." *Information Fusion* 64, 2020.

[5] Wang, Yonghui, Vahid Zarghami, and Suxia Cui. "Fake Face Detection using Local Binary Pattern and Ensemble Modeling." In *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 3917-3921. IEEE, 2021.

[6] Chang, Xu, Jian Wu, Tongfeng Yang, and Guorui Feng. "Deepfake face image detection based on improved VGG convolutional neural network." In *2020 39th chinese control conference (CCC)*, pp. 7252-7256. IEEE, 2020.

[7] Farid, Hany. "Image forgery detection." *IEEE Signal processing magazine* 26, no. 2, 2009.

[8] Hsu, Chih-Chung, Tzu-Yi Hung, Chia-Wen Lin, and Chiou-Ting Hsu. "Video forgery detection using correlation of noise residue." In *2008 IEEE 10th workshop on multimedia signal processing*, pp. 170-174. IEEE, 2008.

[9] Chen, Yi-Lei, and Chiou-Ting Hsu. "Detecting doubly compressed images based on quantization noise model and image restoration." In *2009 IEEE International Workshop on Multimedia Signal Processing*, pp. 1-6. IEEE, 2009.

[10] Guarnera, Luca, Oliver Giudice, and Sebastiano Battiato. "Deepfake detection by analyzing convolutional traces." In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 666-667. 2020.

[11] Rafique, Rimsha, Mariam Nawaz, Hareem Kibriya, and Momina Masood. "DeepFake Detection Using Error Level Analysis and Deep Learning." In *2021 4th International Conference on Computing & Information Sciences (ICCIS)*, pp. 1-4. IEEE, 2021.

[12] Wang, Yonghui, Vahid Zarghami, and Suxia Cui. "Fake Face Detection Using Local Binary Pattern And Ensemble Modeling." In *2021 IEEE International Conference on Image Processing (ICIP)*, pp. 3917-3921. IEEE, 2021.

[13] Taeb, Maryam, and Hongmei Chi. "Comparison of Deepfake Detection Techniques through Deep Learning." *Journal of Cybersecurity and Privacy* 2, no. 1, 2022.

[14] xhlulu. 140k Real and Fake Faces [Dataset]. https://www.kaggle.com/xhlulu/140k-real-and-fake-faces 2020.

[15] Taloba, Ahmed I., Dalia A. Eisa, and Safaa SI Ismail. "A comparative study on using principle component analysis with different text classifiers." *arXiv preprint arXiv:1807.03283*, 2018.

[16] Karamizadeh, Sasan, Shahidan M. Abdullah, Azizah A. Manaf, Mazdak Zamani, and Alireza Hooman. "An overview of principal component analysis." *Journal of Signal and Information Processing* 4, 2020.

[17] Ahmad, Muhammad, Adil Mehmood Khan, Joseph Alexander Brown, Stanislav Protasov, and Asad Masood Khattak. "Gait fingerprinting-based user identification on smartphones." In *2016 International Joint Conference on Neural Networks (IJCNN)*, pp. 3060-3067. IEEE, 2016.

[18] Ahmad, Muhammad, Dr Ihsan Ul Haq, Qaisar Mushtaq, and Muhammad Sohaib. "A new statistical approach for band clustering and band selection using K-means clustering." *Int. J. Eng. Technol* 3, no. 6, 2011

[19] Arjun, V. MANE, Ramesh R. MANZA, and V. KALE Karbhari. "Human face recognition using superior principal component analysis (SPCA)." *International Journal of Computer Theory and Engineering* 2, no. 5, 2010.

[20] Soni, Hritik, Pranjal Arora, and D. Rajeswari. "Malicious Application Detection in Android using Machine Learning." In *2020 International Conference on Communication and Signal Processing (ICCSP)*, pp. 0846-0848. IEEE, 2020.

[21] Rohit, Ramena Venkata Satya, Dhrati Chandrawat, and D. Rajeswari. "Smart Farming Techniques for New Farmers Using Machine Learning." In *Proceedings of 6th International Conference on Recent Trends in Computing*, pp. 207-220. Springer, Singapore, 2021.

[22] Farishta, Kanav Raj, Vivek Kumar Singh, and D. Rajeswari. "XSS attack prevention using machine learning." *World Review of Science, Technology and Sustainable Development* 18, no. 1, 2022.

[23] Agarwal, Harsh, Ankur Singh, and D. Rajeswari. "Deepfake Detection using SVM." In *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pp. 1245-1249. IEEE, 2021.

[24] Han, J., Kamber, M. and Pei, J. “Data mining concepts and techniques third edition,” *The Morgan Kaufmann Series in Data Management Systems*, 5(4), pp. 83–124 . 2011.

[25] Shu, K., Sliva, A., Wang, S., Tang, J. and Liu, H., 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter*, *19*(1), pp.22-36.