Performance Analysis of Face Forgery Recognition and Classification Using Advanced Deep Learning Methods

Babu, Katta Rajesh[1], K. Charan Subhash[1], S. Sumanth[1], Ainala Karthik[1], G. Megana Ram[1], D. Rajendra Prasad[2]

[1]*Department of ECE, KLEF Deemed to be University, Guntur, Andhra Pradesh, India.*
[2]*Depatment of ECE, St. Ann's College of Engineering & Technology, Chirala, Andhra Pradesh, India.*

Abstract: The adoption of web technology has come to be accompanied by a number of worrying security issues, one of which is deep fakes that are now counted among the top visual deceits in the field. The need for identifying such manipulations which is on the rise is the need for stronger methods that can be used to identify such manipulations. This article deals with the usage of fully connected neural networks (FCNN), convolutional neural networks (CNN), and deep convolutional neural networks (DCNN) to determine if a presented facial image is original or fake. In this case, the methods apply the use of the improvements in the feature extracting techniques to catch even the smallest differences in modified materials. Tests conducted on kaggle benchmark datasets depend on that that the methods are the best for it as a solution for safeguarding reliable and efficient forgery detection. The outcome is suggestive of that integration of deep learning methods like CNN, FCNN, and DCNN automated systems has the potential for advancing the struggle against manipulation in media field. As compared with the other models, the CNN is excelling in my testing and it is far better than the rest. More precisely, the CNN is the most perfect while FCNN had its drawbacks in the precision and specificity.

Keywords: Deep fake detection, Face forgery detection, CNN, FCNN, and DCNN.

1. Introduction

Digital forensics is normally at least half if not more of the custom of the entire world, and its importance cannot be underplayed because it pertains to the recovery and evaluation of digital evidence this importance of digital forensics is increased when it comes to legal investigations where the authenticity of applied evidence is vital [1]. A large part of this is proof comes as images and videos, and due to the evolution of technology, pictures have become very editable. An important problem is the robust availability of image editing applications with great capabilities, like those offered by adobe photo shop and light room. Given that the sheer processing power required for massive forms of manipulation gives the forger tremendous power, it has always been concerning how integrity of such image manipulation has always raised serious concerns about the veracity [2].

The need for face forgery detection using CNN, FCNN and DCNN comes from the fact that image and video manipulation technologies are getting more sophisticated and easier to use especially in digital forensics and legal investigations. With Adobe Photoshop and Light room's powerful image processing tools, face forgery has become a big issue. These tools make facial feature modifications easy to do and produce realistic and convincing fake results. So detecting these manipulations is important to have integrity over digital evidence and especially in the judicial domains where image and video authenticity is concerned. This can be addressed by using powerful image analysis capabilities with CNN, FCNN and DCNN. As CNNs can automatically learn hierarchical features from images, they can detect even the slightest alterations in facial structures and visual patterns that are typical of a forgery.

The paper thus aims to investigate their suitability in the contexts of image forgery detection, that is, CNN, FCNN, and DCNN. Specifically, the performance of advanced CNN architectures would be

investigated with utmost attention, including VGG and ResNet, deeper hybrid models, and their ability to detect forgery very precisely.
The article remaining sections are arranged as follows. Rest of the introduction section, section 2 discusses the relevant studies on the detection. In Section 3, we provide a detailed outline of the methods used for comparison. The experimental results and simulations used to assess the efficacy of the deep learning techniques are documented in section 4. In section 5 presented final findings and suggestions for the future.

2. RELATED WORK

This section discusses the recent related work on face forgery classification methods in detail such as
Advances in deep fake and image forgery detection

Chen et al. (2024) planned a submodular subset selection method to improve interpretability by concentrating on fewer, more critical regions in the input data. The method, using CNN and DCNN architectures, ensures the robust detection of minute manipulations in facial images [1]. Yan et al. (2023) proposed the method successfully neutralizes dataset biases by exploring CNN, FCNN, and DCNN models verifying the applicability in actual deployments [2].

Xia et al. (2023) CNN model is proposed, which uses multi-collaboration and multi-supervision mechanisms in sequential deep fake detection. This framework integrates CNN and DCNN layers for feature extraction, focusing on temporal consistency in videos. Model is particularly effective against forged facial movements [3]. Shao et al. (2023), in this study developed methods for robust sequential detection of deep fakes, leveraging DCNN architectures to analyse temporal and spatial features simultaneously. The approach identifies inconsistencies in facial dynamics; the results show significant improvements in precision and recall compared to baseline techniques, especially in high-resolution forgery detection [4].

Neural network models

Liu et al. (2023) suggested stable diffusion methods to improve adversarial transferability in CNN, FCNN, and DCNN models for forgery detection. The authors propose a hybrid approach that combines diffusion-based pre-processing with neural network layers to reduce false negatives. The method proves effective against adversarial attacks and improves detection performance across diverse datasets [5, 6]. Liu et al. (2021) addressed frequency-domain features; spatial-phase shallow learning is presented for facial manipulation detection. The technique relies on CNNs and DCNNs to detect slight inconsistencies in the frequency domain, such as noise artifacts. The authors demonstrate superior accuracy in detecting high-resolution forgeries compared to traditional spatial-domain methods [7].

Adversarial attacks

Wu et al. (2022) established adversarial backdoor attacks and defences benchmarks in neural networks, using CNN and FCNN structures. The authors suggest a multi-layer defence mechanism to recognize and neutralize adversarial inputs. The research has highlighted the need for securely training models and has pinpointed vulnerabilities in state-of-the-art forgery detection models [8-10].

General contributions

Zhao et al. (2021) multi-attentional deep fake detection models are proposed, where CNN and DCNN layers pay attention to region-specific features. This approach effectively identifies subtle artifacts introduced by deep fake algorithms, such as blending inconsistencies. The method is also evaluated on a variety of datasets with high precision and recall [11]. Haliassos et al. (2021) suggested lip-movement-based forgery detection method is proposed, using CNNs for audio-visual synchronization analysis of videos. This method deals with the problem of high-quality forgeries that are missed by traditional pixel-based analysis. The authors achieve important improvements in robustness for a variety of deep fake datasets [12]. M.Cao & Gong (2021) this paper discusses security challenges in deep fake detection systems, focusing on adversarial robustness in CNN and DCNN models. Authors proposed a defence strategy to integrate adversarial training with feature engineering to improve the resilience of the model. Their approach is shown to be effective in reducing false positives in real-world settings [13]. Neekhara et al. (2021) Practical adversarial threats to models of deep fakes detection on CNN and FCNN architectures will be outlined by the authors. A method to optimize feature selection and improve deep learning performance using gated layers is suggested, showing their effectiveness in complex forgery scenarios [14].

Multi-attentional deep fake detection model

Zhao et al. (2021) proposed a multi-attentional deep fake detection model that utilizes CNN and DCNN layers to focus on region-specific features, enabling the identification of subtle artifacts introduced by deep fake algorithms. The approach was evaluated across various datasets, achieving high precision and recall, effectively detecting inconsistencies such as blending artifacts that are often introduced by deep fake techniques [15]. Neekhara et al. (2021) addressed practical adversarial threats to deep fake detection models, particularly for CNN and FCNN architectures. The authors presented a method to optimize feature selection and improve deep learning performance by using gated layers, demonstrating its effectiveness in detecting complex forgeries, even in the presence of adversarial perturbations [16]. Cao & Gong (2021) explored the security challenges in deep fake detection systems, with a focus on adversarial robustness in CNN and DCNN models. They proposed a defence strategy that integrates adversarial training with feature engineering to enhance the model's resilience. Their approach proved effective in reducing false positives in real-world scenarios, improving the robustness of deep fake detection systems [17]. Haliassos et al. (2021) developed a lip-movement-based forgery detection method that uses CNNs for audio-visual synchronization analysis of videos. This method specifically targets high-quality forgeries, which are often missed by traditional pixel-based analysis. By analysing lip movements, the authors were able to detect subtle forgeries with improved robustness across multiple deep fake datasets [18].

Adversarial example generation for deep fake detection

Liang et al. (2021) proposed a method for generating more imperceptible adversarial examples for object detection, which also applies to deep fake detection. In this approach, they utilize CNN and FCNN architectures to improve the detection of adversarial inputs in deep fake models, ensuring that the detection system remains effective even under adversarial attacks, enhancing the robustness of deep fake detection models [19]. Li et al. (2020) presented a "Face X-ray" technique for more general face forgery detection, which integrates CNN and DCNN architectures to analyse facial features at a granular level. The authors demonstrated the effectiveness of this approach in detecting even the most subtle manipulations in faces, providing a robust method for detecting a wide range of forgery techniques across various datasets [20, 21].

3. METHODOLOGY

For face forgery prediction and classification task is evaluated by using the recent deep learning architectures. In this methodology section, the deep learning models (CNN, DCNN, and FCNN) are described in detail given below:

3.1 CNN (Convolutional Neural Network) model

The primary role of CNNs is pivotal in the domain of face forgery detection since it automatically learns and extracts hierarchical features from visual data. Essentially, CNNs have convolutional layers that take input images and process them with filters to capture spatial patterns most are edges, textures, and shapes, often characteristic of manipulations. Pooling layers follow, lowering the dimensionality of the feature maps while retaining only important information, which makes it an efficient model. Classification is then done by full connection layers on images being either authentic or forged depending on the features extracted [22].

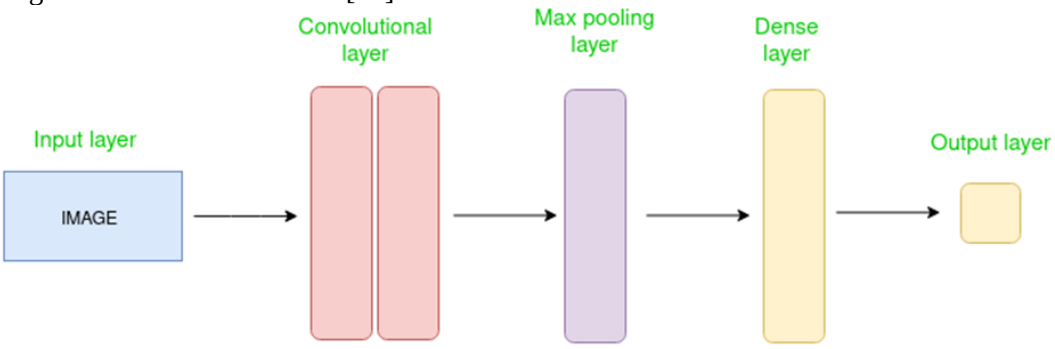


Fig.1.The architecture of CNN

In the Fig.1, the network begins with an input layer that accepts an image as input. The second layer is convolutional layer which uses filters on the input image to get features. This layer typically has several filters, each of which learns to detect different patterns in the image. Third layer is max pooling layer

following the convolutional layer, and this layer is applied to down sample the feature maps, thus decreasing their spatial size and preserving significant information. This would decrease the computational cost and improve the network's resistance to small variations in the input image. Fourth layer is dense layer and this one is also referred to as fully connected layer. It is actually the layer that makes a prediction and taking the output from all other layers and using it for decision-making. For example how to classify the image in question. The last layer is the output layer, which gives the final prediction or classification output [23].

3.2 DCNN (Deep Convolutional Neural Network) model

A DCNN is an advanced form of a convolutional neural network, designed to handle more complex tasks by using a deeper architecture. The main characteristic of DCNNs is that they contain multiple layers stacked one after another, which increases the depth of the network compared to traditional CNNs. This enables the model to learn hierarchical features from raw input data, like images or time series, and hence improve the accuracy of the predictions [24].

In DCNN, deeper architectures enable the model to capture complex patterns and learn abstract representations from the data. The more the network is deepened, the better it can identify higher-level features, such as objects in images or more complex sequences in time series. This is achieved by non-linear activation functions that enable the network to learn complex mappings from input to output. Another important aspect is the use of dropout, batch normalization, and other techniques in avoiding overfitting and generalizing the model well for unseen data.

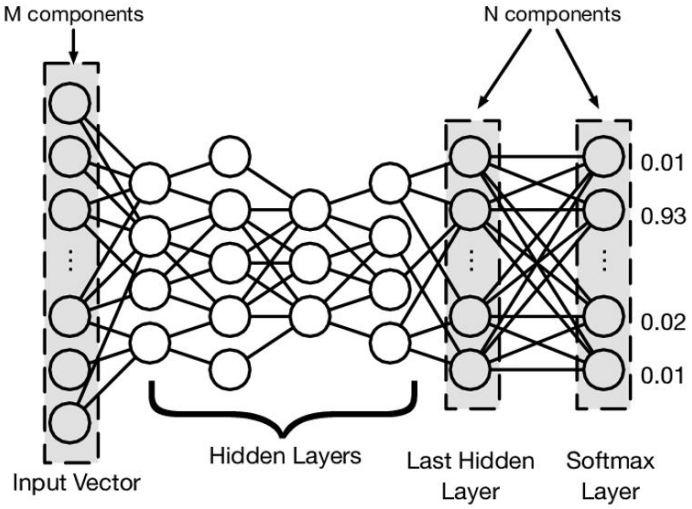


Fig.2.The model of DCNN

In Fig.2 the first layer is input vector which shows a feedforward neural network with multiple hidden layers. Such a network processes the information in one direction, namely, from the input layer to the output layer. The input layer is the layer that accepts the input data, as the vector X. In this case, there are M elements in the input vector. The hidden layers take in the input data, extract features and pass them through. There are several hidden layers in this image, all with different neurons [25].

The last hidden layer is before the output layer and it is referred to as Z(X) because it displays the representation of the input. The soft max layer is the output layer of the network. It receives the output from the last hidden layer and generates a probability distribution over the possible output classes. Here, there are N elements of the output vector. Connections: Weighted Links are the connections that display the parameters of the neural network. The link on each connection indicates how strong the connection is between two neurons. The neurons within a layer are not necessarily identical. Neuron connections are weighted, and the weights are learned during training [26].

The softmax layer has output probabilities sum up to 1, which make it good for classification problems. This type of neural network architecture is useful for the tasks like image classification, object detection as well as natural language processing.

3.3 FCNN (Fully Connected Neural Network) model

The FCNN architecture is a basic model in deep learning, mainly used for a wide array of computational tasks. This type of architecture consists of several layers of nodes, also called neurons, where each neuron in a given layer is connected with every neuron in the previous and following layers. This extensive interconnectivity enables FCNNs to learn and capture complex patterns from data very

effectively, making them highly applicable to various applications, such as image and speech recognition, natural language processing, and more.

A typical modular structure of an FCNN includes an input layer, multiple hidden layers, and an output layer. This is where the raw data is fed to the input layer, which further passes it to the hidden layers using activation functions such as rectified linear units or sigmoid functions. Activation functions introduce non-linearity in the model and allow the network to learn more complex representations of the input data. Finally, the output layer makes the final predictions based on the transformations that occurred in the preceding layers [27].

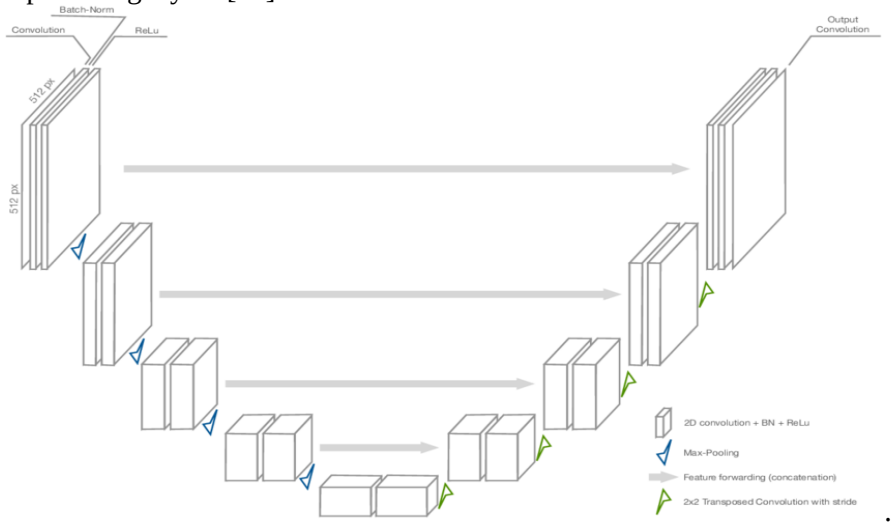


Fig.3.The architecture of FCNN model

In Fig.3 the first input convolution layer is a 512 ×5 12 pixel image, which is the raw data the network processes. It applies filters to the input image to extract features. Then, there's a Batch Normalization layer, which helps stabilize the training process. Next layer is a ReLU activation function is applied to introduce non-linearity. This pattern of convolution-batch norm-ReLU repeats 20 times, forming a deep convolutional network [28]. The middle layer is pooling layer after the convolutional blocks, a max-pooling layer is applied in order to down sample the feature maps, reducing their spatial dimensions but keeping the important information. Next layer is feature forwarding (Concatenation), it combines feature maps coming from different stages of the network. This way, the network is able to learn more complex representations. At last layer is transposed convolutional layer is a 2 × 2 transposed convolutional layer with stride is used to up sample the feature maps, increasing their spatial dimensions. The last layer output layer provides outputs processed image, which would possibly be a reconstructed or generated one [29].

3.4 Performance metrics

There are various metrics are used to measure the performance of each method to detect the face forgery detection. Some important measures such as accuracy, precision, specificity, recall and F1-Score used to classify the variation between the real and fake images. These are also called as classification metrics and derived from the 2-D matrix i.e., confusion matrix shown in Fig.4. This compromise between the actual values and predicted values that values are calculated by TP (true positive), TN (true negative), FP (false positive), and FN (false negative) [30].

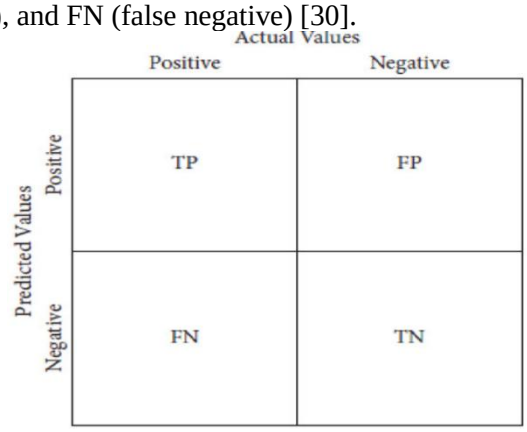


Fig.4.The 2-D confusion matrix representation

Classification metrics: Some classification mathematical formulas are given below:
Accuracy: Accuracy represents the overall correctness of the model across all predictions.

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (1)$$

Precision: It indicates the proportion of correctly identified positive instances out of all instances predicted as positive.

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (2)$$

Specificity: It calculates the proportion of actual negative instances correctly identified.

$$\text{Specificity} = \frac{\text{TN}}{\text{TN+FP}} \quad (3)$$

Recall (Sensitivity): This one measures the proportion of actual positive instances correctly identified.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (4)$$

F1-Score: It is the harmonic mean of Precision and Recall, balancing both metrics.

$$\text{F1} - \text{Score} = 2 \times \frac{\text{recall} \times \text{precision}}{\text{recall+precision}} \quad (5)$$

4. RESULTS & DISCUSSION

In this work, there are several steps involved. Initially, the dataset was pre-processed. For validation, we have utilized holdout validation. The deep learning algorithms such as CNN, FCNN, and DCNN were utilized to train the images.
Experimental setup: The intended architectures were deployed on Google Co-lab with an 11th Gen Intel® Core™ i3-1115G4 processor that had a 3.00 GHz speed and 8GB of RAM.
Dataset: The face forgery detection dataset is consisting of 1000 faces images of both real and fake images downloaded from public available kaggle benchmark datasets, later which can analyze using the deep leaning method. The dataset was divided into 80% training, 10% testing, and 10% validation. The sample fake and real images displayed in the Fig.6 and Fig.7

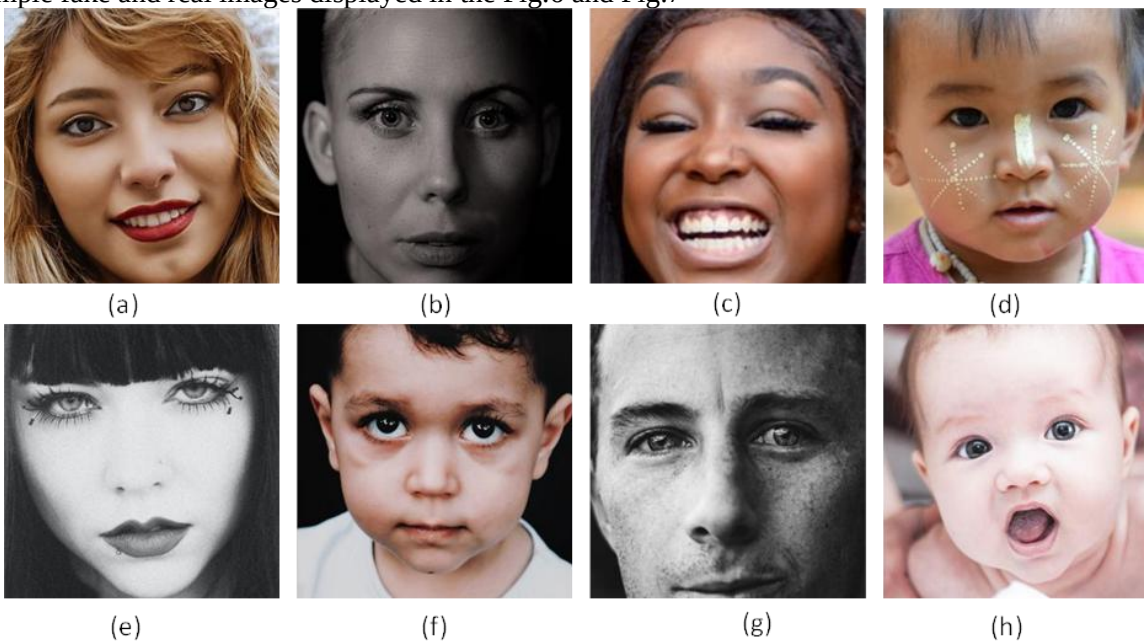


Fig.6.The sample fake faces images from (a) to (h).

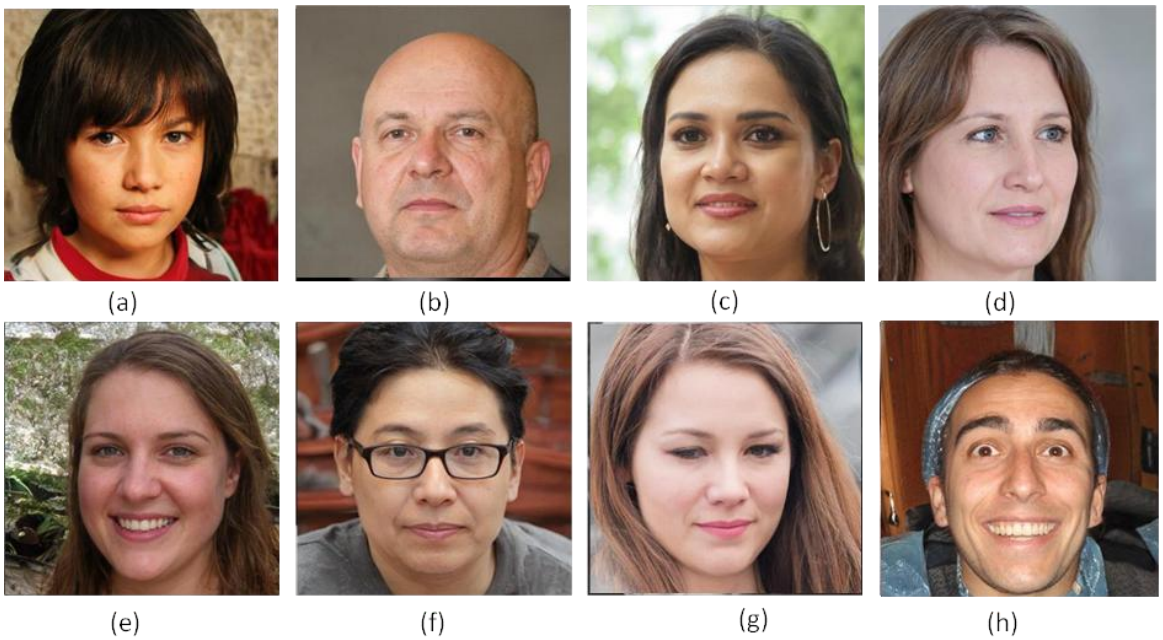


Fig.7.The real faces from (a) to (h).

For validation of our results, we have calculated performance metrics of precision, recall, specificity, and accuracy, and F1-Score. These are computed based on confusion matrix that is derived based on prediction results on the combination data that is both testing and training sets.

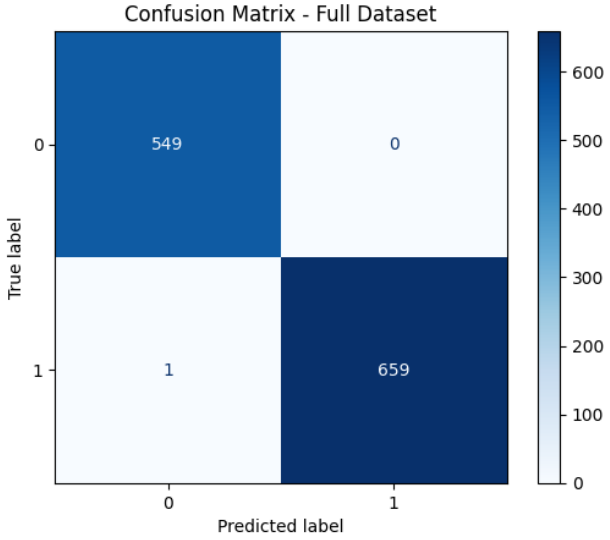
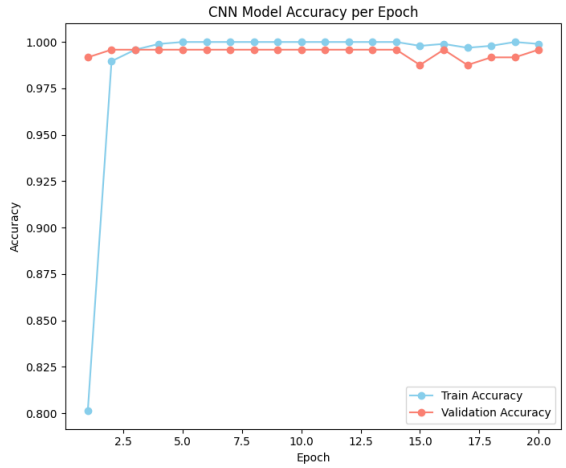


Fig.8. Confusion matrix of CNN model

In Fig.8, shows the confusion matrix based on a granular, detailed analysis of classifying two categories of instances. With 549 true positives and an equally impressive 659 true negatives, one can thus see that the performance of these predictions is excellent. However, that there are no false positives is especially remarkable-this would only mean that the system has learnt to avoid Type I errors-situations where instances are incorrectly designated as belonging to the positive class. While a single false negative is observed, there is only a minor potential for Type II errors (incorrectly classifying positive instances as negative), which does not significantly impact the model's overall strong performance.



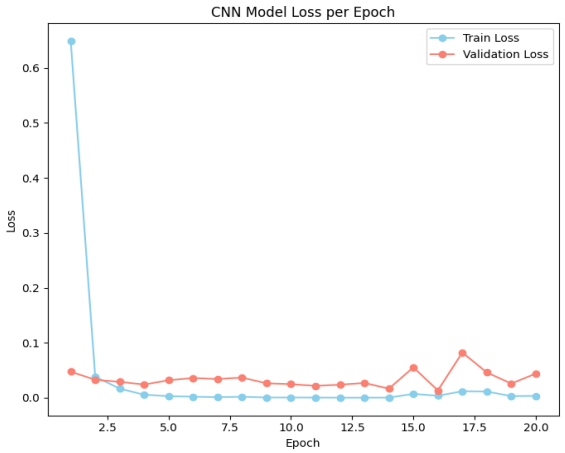


Fig.9. Train and validation accuracy of CNN Fig.10. Graph of train and validation loss

As per the Fig.9 at around the 10th epoch, the validation accuracy starts to plateau and then decline. This means that the model has started to over fit the training data. Overfitting is a phenomenon in which the model learns the training data too well, memorizing the details rather than learning generalizable patterns. This causes the performance on new, unseen data to decline. Provided in the graph is the training and validation accuracy of a CNN model over 20 epochs. The blue line denotes training accuracy, which indicates the model's performance on data with which it was trained. One expects training accuracy to increase steadily since it shows how well the model learned patterns on the training data. But the red curve, which corresponds to the validation accuracy, shows a different behaviour. For a short period, even the validation accuracy improves and shows generalization to unseen data. However, after reaching its peak somewhere around the 10th epoch, it stabilizes and even starts decreasing. This kind of behaviour is called overfitting between training and validation accuracy.

In Fig.10 initially, the training loss starts from a higher value (~0.6) and decreases rapidly in the first two iterations, indicating rapid learning. The validation loss decreases, indicating that the model can generalize well to the unseen data. However, between iterations 3 and 10, the improvement slows down, and the training loss continues to decrease, but very slowly. During this period, the validation loss remains stable, varying slightly but not significantly, indicating that the model has maintained its ability to generalize. From iterations 11 to 20, the training loss continues to decrease, but at a slower rate, indicating that the model is approaching its learning limit. However, validation losses began to decrease, and sometimes even increase, indicating the onset of saturation as the model began to focus on the training data and had difficulty generalizing to new data.

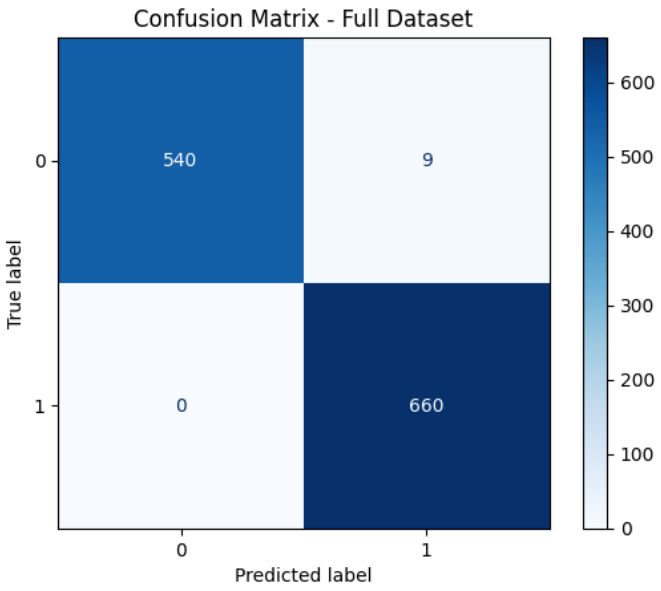
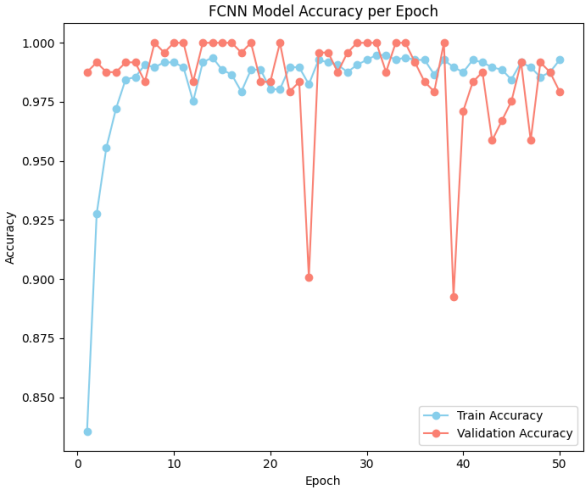


Fig.11. Confusion matrix of FCNN model

In Fig.11 the confusion matrix provides an in-depth performance evaluation of the classification model for the entire dataset. For true positives, it has a count of 540, while for true negatives; the count is 660, which shows the model can predict both positive and negative instances with high accuracy. Only 9 false positives confirm that the model is performing its task well to avoid Type I errors, which are classifying negative instances as positive instances. In addition, no false negatives are found, and hence the model captures all the true positives. In the meantime, it is also worthwhile to note that data imbalance could play a role in the performance. That is, if the instances are mostly negative, it might affect the performance and one should interpret the findings based on this fact as well.

In Fig.12, initially, training accuracy starts out at a low level but increases rapidly over the first two epochs, indicating that the model is learning quickly and making great progress. Confirmed accuracy increases, indicating good generalization of unseen data. At intermediate levels, training accuracy increases, but at low rates the model is still able to learn, albeit at a slower rate. Confirmed accuracy varies during this period, increasing and decreasing, showing inconsistent progress across the model. In other systems, training accuracy improves, but in most enterprises the models tend to be too simple. The accuracy of the predictions starts to drop significantly, indicating that the model does not remember the training data and cannot accurately predict new unseen data.



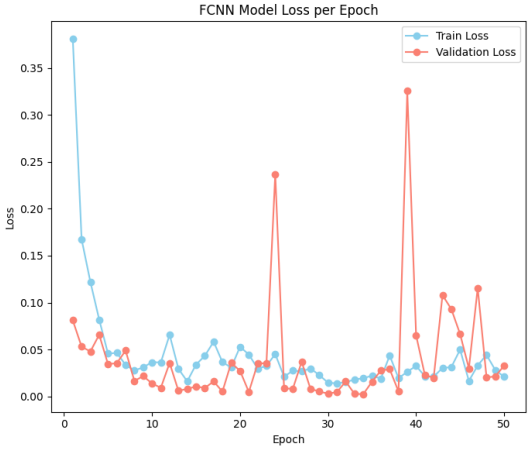


Fig.12. Train and validation accuracy of FCNN Fig.13.Train and validation loss of FCNN

In Fig.13 the first part, the training loss begins with a very high value and drops sharply in the first two phases, showing that the model learns fast and minimizes the application error. The confidence interval also drops, showing that the model is a good fit for the unobserved data. In the middle range, the training loss drops, but slowly, showing a slow learning curve. The loss in support starts changing, at times rising and then falling, signalling that the overall model is unreliable. In later iterations, training loss changes vastly, at times rising and sometimes falling, suggesting that the process of learning is unstable and leading to overfitting. The confidence intervals are also varying, with certain higher values reflecting that the model is well-fitting the training data, and overall performance on unseen data is deteriorating.

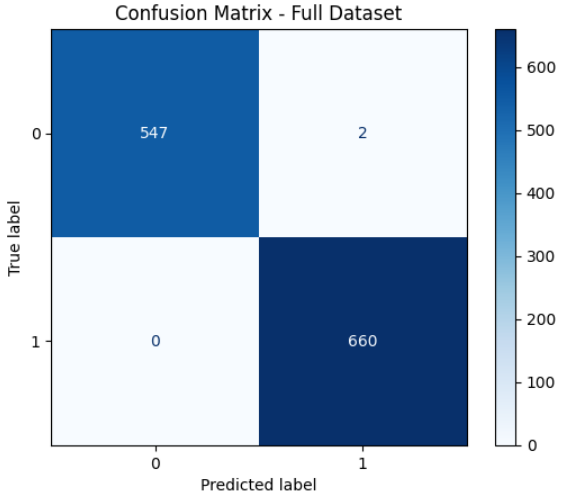
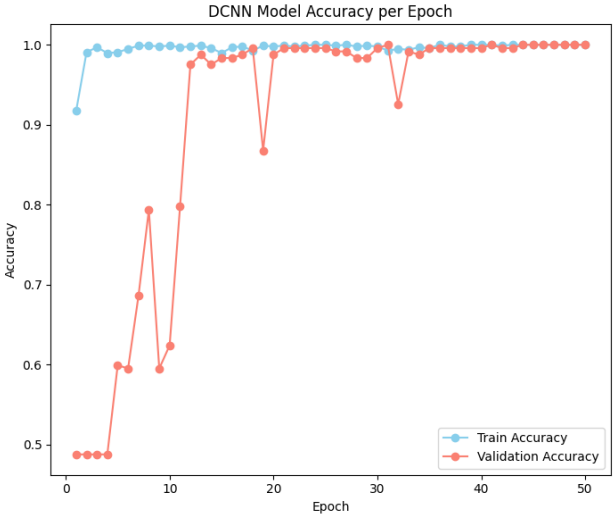


Fig.14. Confusion matrix of DCNN model

In the Fig.14, the confusion matrix presented here gives a comprehensive view of how the model classifies instances into two categories. With 547 true positives and 660 true negatives, the model has high accuracy for predicting both positive and negative instances. With only 2 false positives, the model is very effective in avoiding Type I errors, where the negative instances are wrongly classified as positive. Moreover, there is no false negative, which indicates that the model has identified all the true positives. But the dataset could be imbalanced, and it has more negative instances. This might influence the performance of the model and the interpretation of the results.



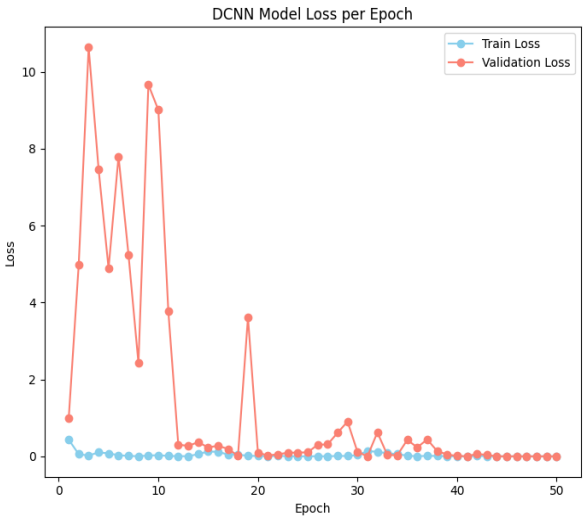


Fig.16.Train and validation accuracy of DCNN Fig 17.Train and validation loss of DCNN

In Fig.16 the first stage, the training accuracy starts low, but increases rapidly over the first two cycles, indicating that the model is learning quickly and efficiently. The validation accuracy also increases during this period, indicating a good fit to the unseen data. In the second stage, training continues, but at a slower rate, indicating gradual learning. The real-time changes and the increase and decrease times indicate a lack of consistency in the overall model's capabilities. In the next stage, the training accuracy continues to increase, but with the observed changes, indicating the potential for better performance. The accuracy begins to decline, indicating that the model has become limited to the training data and is losing its general ability to new data.

In the fig 17 the first stage, the learning loss starts with a significant value, but decreases rapidly in the first two stages, indicating rapid learning and a decrease in error. The validation loss is also reduced, which has a general effect on the observed data. In the intermediate stage, the learning loss shows large fluctuations, spikes and dips, indicating the instability of the learning process. The validation loss also oscillates, with large spikes at some points, suggesting overfitting due to the previous model of the data. In the later stages, the learning loss continues to fluctuate, sometimes increasing, and the negative reinforcement shows a regular pattern with some increases. This inconsistency and increased acceptance loss further indicate that the model is overfitting, affecting its ability to adapt to new information.

Table 1 shows the performance metrics of accuracy, precision, recall, specificity, along the advanced deep leaning models such as CNN, FCNN and DCNN.

Table 1: Comparison between metrics and deep learning models

Metrics	CNN	DCNN	FCNN
Precision	1.0000	0.9919	0.9466
Recall(sensitivity)	0.9985	0.9919	1.0000
Specificity	1.0000	0.9915	0.9407
F1-Score	0.9992	0.9919	0.9725
Accuracy	0.9992	0.9917	0.9711

The precision of the CNN method is the highest at 1.0000, suggesting the model correctly labelled all the positive instances without false positives. The precision of the DCNN and FCNN methods are slightly lower, at 0.9919 and 0.9466, respectively.

Recall (sensitivity) is the model's capacity to correctly classify all positive examples. The method FCNN has a perfect recall of 1.0000, so it classified all the positive cases. Both CNN and DCNN methods have similar recalls at 0.9985 and 0.9919, respectively.

Specificity reflects how well the model can correctly identify negative instances, and the CNN model shows perfect specificity at 1.0000. The DCNN method is slightly lower, at 0.9915, while the FCNN model has the lowest specificity at 0.9407, meaning it misclassified more negative cases than the other two models.
CNN again outperforms with an F1 score of 0.9992. The DCNN method follows closely at 0.9919, while FCNN has a lower F1 score of 0.9725, reflecting its relatively lower precision.
Accuracy is the overall correctness of the model. CNN has obtained a maximum accuracy of 0.9992, DCNN with 0.9917, and FCNN is having the lowest accuracy of 0.9711; it shows that CNN did well in all the tasks compared to the other models for classification.
In short, CNN performs better than the others regarding precision, specificity, F1-score, and accuracy while FCNN shows performance in recall but seems weak in precision and specificity. DCNN seems balanced but is not able to outperform CNN in any of the evaluation metrics.

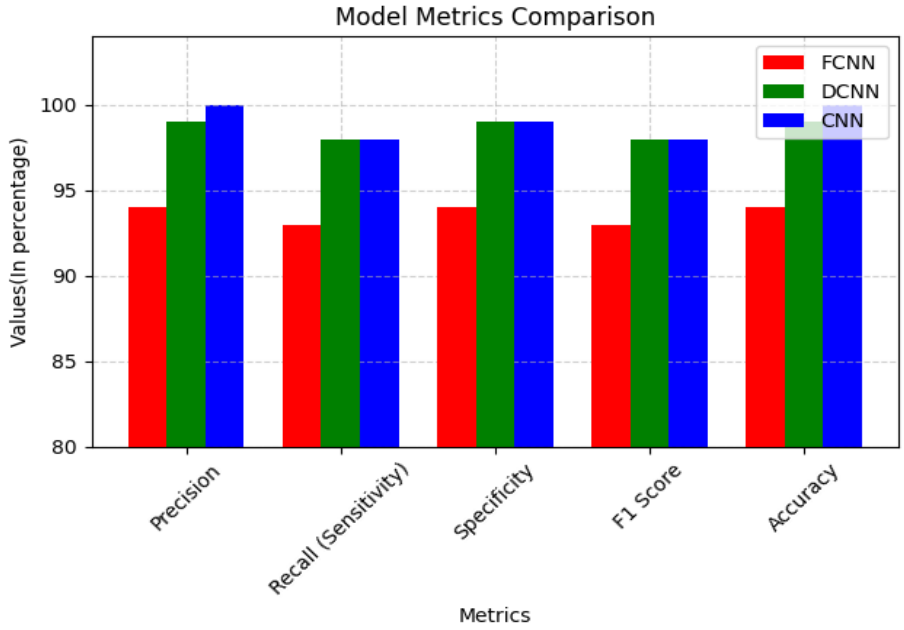


Fig.18. Comparison between values of different methods and metrics

In the Fig.18 the Bar Graph Precision: All of the models, CNN, DCNN, and FCNN display high precision values, which means positive predictions. CNN has highest precision which shows that its true positives are the largest.
Recall (Sensitivity): All of the models clearly show great recall values, suggesting that these models are very good at actually identifying the true positive cases CNN once again is seen to hold the highest value of recall
Specificity: All models have good specificity values, meaning that they are performing well with true negative cases. CNN is slightly ahead of the other two models on this score.
The F1 score, which is a harmonic mean of precision and recall, is high for all the models. CNN has scored the highest F1 score which indicates a good balance between precision and recall.
Accuracy all three models have shown high accuracy, indicating that they could make correct predictions in the majority of cases. Among the three models, CNN has the highest accuracy.

5. CONCLUSION

According to the comparative analysis of fake forgery recognition and classification using the CNN, DCNN, and FCNN methods, the CNN exhibits the highest performance in all the metrics. It shows perfect precision and specificity, near-perfect recall, F1-score, and accuracy. The results indicate that CNN has the ability to minimize false positives and false negatives. DCNN closely follows with consistent performance, having an accuracy of and balanced precision and recall. However, it is slightly low in specificity compared to CNN. FCNN has perfect recall at but is low in precision and specificity, which lowers the accuracy. Therefore, overall the CNN is the best model as compared to the other two models for fake face prediction by achieving maximum accuracy and reliability.

Further possibilities can be the introduction of new architectures, such as transformer-based models or hybrid networks based on the combination of CNNs and attention mechanisms that improve model performance. The generalization of the model would improve by expanding the dataset with more diverse face images, such as different lighting, angles, and ethnicities.
Data Availability: https://www.kaggle.com/datasets/ciplab/real-and-fake-face-detection
Complicit of interest: No

References

1. Chen et al. (2024)Ruoyu Chen, Hua Zhang, Siyuan Liang, Jingzhi Li, and Xiaochun Cao.Less is more: Fewer interpretable region via submodular subset selection.arXiv preprint arXiv:2402.09164, 2024.
2. Yan et al. (2023)Zhiyuan Yan, Yong Zhang, Yanbo Fan, and Baoyuan Wu.Ucf: Uncovering common features for generalizable deepfake detection.arXiv preprint arXiv:2304.13949, 2023.
3. Xia et al. (2023)Ruiyang Xia, Decheng Liu, Jie Li, Lin Yuan, Nannan Wang, and Xinbo Gao.Mmnet: Multi-collaboration and multi-supervision network for sequential deepfake detection.arXiv preprint arXiv:2307.02733, 2023.
4. Shao et al. (2023)Rui Shao, Tianxing Wu, and Ziwei Liu.Robust sequential deepfake detection.arXiv preprint arXiv:2309.14991, 2023.
5. Liu et al. (2023)Jiayang Liu, Siyu Zhu, Siyuan Liang, Jie Zhang, Han Fang, Weiming Zhang, and Ee-Chien Chang.Improving adversarial transferability by stable diffusion.arXiv preprint arXiv:2311.11017, 2023.
6. Liu et al. (2023)Aishan Liu, Xinwei Zhang, Yisong Xiao, Yuguang Zhou, Siyuan Liang, Jiakai Wang, Xianglong Liu, Xiaochun Cao, and Dacheng Tao.Pre-trained trojan attacks for visual recognition.arXiv preprint arXiv:2312.15172, 2023.
7. Liu et al. (2023)Aishan Liu, Shiyu Tang, Xinyun Chen, Lei Huang, Haotong Qin, Xianglong Liu, and Dacheng Tao.Towards defending multiple lp-norm bounded adversarial perturbations via gated batch normalization.International Journal of Computer Vision, 2023b.
8. Liu et al. (2023)Aishan Liu, Jun Guo, Jiakai Wang, Siyuan Liang, Renshuai Tao, Wenbo Zhou, Cong Liu, Xianglong Liu, and Dacheng Tao.X-adv: Physical adversarial object attacks against x-ray prohibited item detection.In USENIX Security Symposium, 2023a.
9. Wu et al. (2022)Baoyuan Wu, Hongrui Chen, Mingda Zhang, Zihao Zhu, Shaokui Wei, Danni Yuan, and Chao Shen.Backdoorbench: A comprehensive benchmark of backdoor learning.Advances in Neural Information Processing Systems, 35:10546–10559, 2022.
10. Wang et al. (2022)Yuhang Wang, Huafeng Shi, Rui Min, Ruijia Wu, Siyuan Liang, Yichao Wu, Ding Liang, and Aishan Liu.Adaptive perturbation generation for multiple backdoors detection.arXiv preprint arXiv:2209.05244, 2022b.
11. Wang et al. (2022)Tong Wang, Yuan Yao, Feng Xu, Shengwei An, Hanghang Tong, and Ting Wang.An invisible black-box backdoor attack through frequency domain.In European Conference on Computer Vision, pp. 396–413. Springer, 2022a.
12. Shiohara & Yamasaki (2022)Kaede Shiohara and Toshihiko Yamasaki.Detecting deepfakes with self-blended images.In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 18720–18729, 2022.
13. Shao et al. (2022)Rui Shao, Tianxing Wu, and Ziwei Liu.Detecting and recovering sequential deepfake manipulation.In European Conference on Computer Vision, pp. 712–728. Springer, 2022.
14. Liang et al. Parallel rectangle flip attack: A query-based black-box attack against object detection.arXiv preprint arXiv:2201.08970, 2022c.
15. Zhao et al. (2021)Hanqing Zhao, Wenbo Zhou, Dongdong Chen, Tianyi Wei, Weiming Zhang, and Nenghai Yu.Multi-attentional deepfake detection.In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 2185–2194, 2021.
16. Neekhara et al. Ferrer.Adversarial threats to deepfake detection: A practical perspective.In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 923–932, 2021.
17. Liu et al. (2021)Honggu Liu, Xiaodan Li, Wenbo Zhou, Yuefeng Chen, Yuan He, Hui Xue, Weiming Zhang, and Nenghai Yu.Spatial-phase shallow learning: rethinking face forgery detection in frequency domain.In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 772–781, 2021b.
18. Cao & Gong (2021)Xiaoyu Cao and Neil Zhenqiang Gong.Understanding the security of deepfake detection.In International Conference on Digital Forensics and Cyber Crime, pp. 360–378. Springer, 2021.
19. Haliassos et al. (2021)Alexandros Haliassos, Konstantinos Vougioukas, Stavros Petridis, and Maja Pantic.Lips don't lie: A generalisable and robust approach to face forgery detection.In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 5039–5049, 2021.

20. Liang et al. (2021)Siyuan Liang, Xingxing Wei, and Xiaochun Cao.Generate more imperceptible adversarial examples for object detection.In ICML 2021 Workshop on Adversarial Machine Learning, 2021.
21. Li, Lingzhi, et al. "Face x-ray for more general face forgery detection." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.
22. Velliangira, S., and J. Premalata. "A novel forgery detection in image frames of the videos using enhanced convolutional neural network in face images." Computer Modeling in Engineering & Sciences 125.2 (2020): 625-645.
23. Zamir, Muhammad, et al. "Face detection & recognition from images & videos based on CNN & Raspberry Pi." Computation 10.9 (2022): 148.
24. Ferrari, Claudio, et al. "Investigating nuisances in DCNN-based face recognition." IEEE Transactions on Image Processing 27.11 (2018): 5638-5651.
25. Reis, Paulo Max Gil Innocencio, and Rafael Oliveira Ribeiro. "A forensic evaluation method for DeepFake detection using DCNN-based facial similarity scores." Forensic Science International 358 (2024): 111747.
26. Reis, Paulo Max Gil Innocencio, and Rafael Oliveira Ribeiro. "A forensic evaluation method for DeepFake detection using DCNN-based facial similarity scores." Forensic Science International 358 (2024): 111747.
27. Basha, SH Shabbeer, et al. "Impact of fully connected layers on performance of convolutional neural networks for image classification." Neurocomputing 378 (2020): 112-119.
28. Nagpal, Chaitanya, and Shiv Ram Dubey. "A performance evaluation of convolutional neural networks for face anti spoofing." 2019 international joint conference on neural networks (IJCNN). IEEE, 2019.
29. Kasar, Manisha M., Debnath Bhattacharyya, and T. H. Kim. "Face recognition using neural network: a review." International Journal of Security and Its Applications 10.3 (2016): 81-100.
30. Khodabakhsh, Ali, et al. "Fake face detection methods: Can they be generalized?." 2018 international conference of the biometrics special interest group (BIOSIG). IEEE, 2018.