JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

IMAGE FORGERY DETECTION USING MACHINE LEARNING ALGORITHMS

[1]Tanush Shekharappa Gouda, [2]M Ravishankar, [3]Dinesha H A
[1]Research Scholar
[1]Computer Science Engineering Department
[1]VVIET Mysuru, Affiliation to VTU Belagavi, INDIA

Abstract: Convolutional Neural Networks (CNNs) are used for the image processing applications such as image segmentation, classification, detection, image context and retrieval related tasks. The topology of CNN typically consists of different filters and fully connected layers with non linear processing units for image classification and recognition. It helps to exploit the spatial or temporal correlation in the underlying data. In this paper, the image forgery detection through experiments on different CNN or DL architectures and ML algorithms.

Keywords: **Image Forgery Detection, Machine Learning Algorithms, Deep Learning, CNN, SVM, ANN**

I. INTRODUCTION

Image forgery detection is a crucial task in the field of image processing and computer vision. With the widespread use of digital images and easy access to image editing tools, the need for reliable and efficient methods to detect image forgery has become more important than ever.

One of the approaches to detect image forgery is through a hybrid technique that combines different methods for improved accuracy. A novel hybrid technique could involve combining techniques such as:

1. ***Digital image forensics***: This involves analyzing the statistical properties of digital images to detect any tampering. Techniques such as detecting inconsistencies in noise patterns, detecting duplication or splicing of image regions, and identifying traces of manipulation in image metadata can be used.

2. ***Deep learning***: Deep learning methods such as Convolution Neural Networks (CNNs) have shown great promise in detecting image forgery. They can be trained on large datasets of authentic and manipulated images to learn features that can help distinguish between the two.

3. ***Hashing techniques***: Hashing techniques can be used to detect image tampering by comparing the hash values of different image parts. If the hash values of two image parts differ, it indicates that they have been tampered with.

4. ***Watermarking***: Watermarking can be used to embed a unique identifier in an image, which can later be used to verify its authenticity. This can be particularly useful in cases where the image has been resized or compressed.

By combining these techniques, a novel hybrid technique can provide a more robust and accurate approach to detect image forgery. The specific implementation of the technique will depend on the nature of the images being analyzed and the types of forgery being targeted in fig [1].

Image forgery detection is the process of identifying any manipulation or tampering of digital images. Image processing techniques can be used to detect various types of image forgery, such as copy-move forgery, splice forgery, and removal forgery.

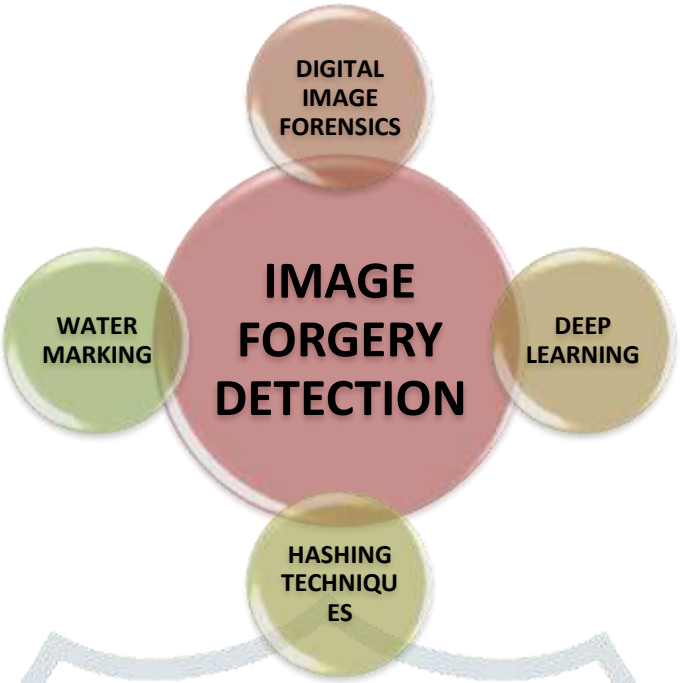

Figure 1. Image Forgery Detection

II. RELATED WORK

Here are some common Image Processing Techniques used for Image Forgery Detection.

1. ***Copy-Move Forgery Detection***: Copy-move forgery is a common type of image forgery where a part of an image is duplicated and pasted onto another part of the same image. This can be detected using techniques such as block matching and key point-based methods.

2. ***Splice Forgery Detection***: Splice forgery is another type of image forgery where parts of two or more different images are combined to create a new image. This can be detected using techniques such as JPEG compression analysis, sensor pattern noise analysis, and frequency-based methods.

3. ***Removal Forgery Detection***: Removal forgery involves removing or hiding certain parts of an image. This can be detected using techniques such as image in painting analysis and inconsistency detection.

4. ***Metadata Analysis***: Metadata analysis involves examining the metadata of an image file to determine if it has been manipulated or altered. This can include information such as the date and time the image was taken, the camera settings used, and the location.

Overall, image processing techniques can be used to detect various types of image forgery by analyzing different aspects of the image such as the pixel values, compression artifacts, and metadata. However, it is important to note that these techniques may not always be 100% accurate and may require manual verification.

Image forgery detection is an important application of image processing and machine learning. Image forgery can occur in various forms, such as copy-paste, splicing, and cloning, where parts of an image are duplicated or moved to a different location to create a new image. The objective of image forgery detection is to identify whether an image has been manipulated or not.

Machine learning algorithms can be used to automatically detect image forgery by learning patterns in the image data. Here are some common approaches for using machine learning in image forgery detection:

1. ***Feature-based approach***: In this approach, various features such as color histograms, texture, and edge information are extracted from the image. These features are then used as inputs to a machine learning algorithm, which learns to classify images as genuine or forged based on these features.

2. ***Deep learning approach***: Deep learning algorithms such as convolutional neural networks (CNNs) have shown remarkable success in image classification tasks. In image forgery detection, CNNs can be trained on large datasets of genuine and forged images to automatically learn features that are useful for detecting image manipulation.

3. ***Hybrid approach***: A combination of feature-based and deep learning approaches can be used to improve the accuracy of image forgery detection. For example, a deep neural network can be trained to extract features from the image, which are then fed into a machine learning algorithm for classification.

Overall, machine learning is a powerful tool for detecting image forgery, and its effectiveness can be further improved by combining it with traditional image processing techniques. Deep learning is a subfield of machine learning that utilizes neural

networks to learn and model complex patterns in data. CNNs are a type of deep learning algorithm that is particularly well-suited for image recognition and computer vision tasks [4].

CNNs are designed to recognize patterns in visual data by employing a series of convolution layers. Each convolution layer consists of a set of learnable filters that convolve over the input image, producing a set of feature maps. These feature maps are then passed through a non-linear activation function, such as a ReLU (Rectified Linear Unit), to introduce non-linearity into the model.

The output from the convolution layers is then passed through one or more fully connected layers, which are similar to the layers in a traditional neural network. The final layer in the network is typically a softmax layer, which produces a probability distribution over the possible classes that the input image belongs to.

Training a CNN involves adjusting the weights of the filters in the convolution layers and the weights in the fully connected layers through a process called back propagation. During training, the network is presented with a large set of labelled examples, and the weights are adjusted in order to minimize a loss function that measures the difference between the predicted outputs of the network and the true labels.

CNNs have been successfully applied to a wide range of image recognition tasks, such as object detection, face recognition, and image segmentation. They have also been used for tasks such as natural language processing and speech recognition, where the input data can be represented as a sequence of vectors [6].

Image forensics is the process of analyzing digital images to determine their authenticity and integrity. This includes identifying whether an image has been manipulated or edited in any way, detecting any tampering or alteration, and determining the source and origin of the image [8].

The field of image forensics combines techniques from various disciplines such as digital signal processing, computer vision, cryptography, and machine learning. Some common methods used in image forensics include:

1. ***Digital watermarking***: This involves embedding a unique digital signature or watermark into an image to identify its origin or ownership.

2. ***Error level analysis***: This technique analyzes the compression artifacts in an image to detect any inconsistencies or discrepancies that may indicate tampering.

3. ***Metadata analysis***: The metadata of an image can provide information such as the date, time, and location of the image, which can be used to verify its authenticity.

4. ***Duplicate image detection***: This technique involves comparing an image to a database of known images to determine if it has been duplicated or altered.

5. ***Image forgery detection***: This method uses machine learning algorithms to detect if an image has been digitally manipulated or forged.

Image forensics is important in many fields such as law enforcement, journalism, and digital art, as it helps to ensure the accuracy and authenticity of images used as evidence or for publication deployed in fig 2.

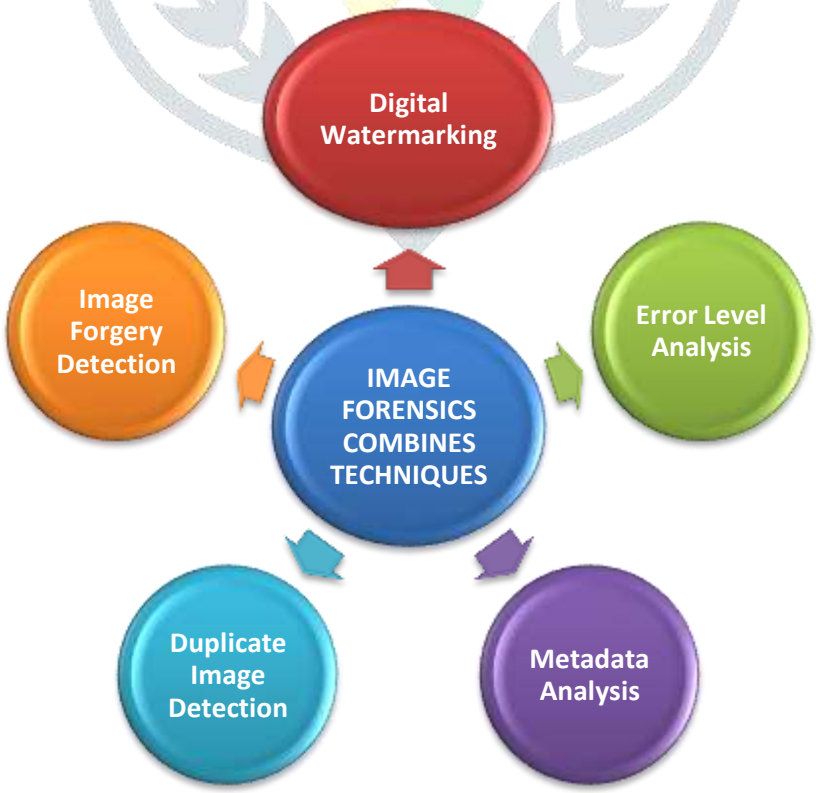

Figure 2. Image Forensics Combines Techniques from Various Disciplines

III. RESEARCH METHODOLOGY, RESULTS AND DISCUSSION

ERROR LEVEL ANALYSIS

Error Level Analysis (ELA) is a technique used in image forensics to detect digital image manipulation or compression artifacts. It involves calculating the difference between the original image and a recompressed or manipulated version of the same image, and then visualizing this difference as an image [10].

Here are the steps to perform an Error Level Analysis:

1. Take the original image and make a copy of it.
2. Re-save the copy with a lossy compression algorithm (such as JPEG) at a high-quality level.
3. Calculate the difference between the original and the compressed image.
4. Convert the difference image to grayscale.
5. Adjust the contrast of the difference image to make the changes more visible.

The resulting image is the Error Level Analysis image, which can help identify areas of the image that have been manipulated or compressed more heavily than others. In general, areas with high levels of manipulation or compression will appear as regions of high contrast in the ELA image.

ELA is not foolproof, and there are some limitations to its effectiveness. For example, it may not be able to detect subtle image manipulations or certain types of compression. Additionally, the technique may produce false positives in images that have been legitimately recompressed for other reasons, such as resizing or cropping.

Overall, ELA can be a useful tool in image forensics, but it should be used in conjunction with other techniques and approaches for a more comprehensive analysis.

Forensic similarity analysis is the process of comparing two or more digital images to determine if they are similar or identical. This process is commonly used in digital forensics investigations to determine if images have been altered or manipulated [1].

As described in fig [3], there are several methods for performing forensic similarity analysis on digital images, including:

1. ***Hash-based analysis***: This involves calculating a hash value for each image and comparing them to see if they are identical. A hash value is a unique digital fingerprint of an image that is calculated using a mathematical algorithm.

2. ***Pixel-based analysis***: This involves comparing the pixels of the images to see if they are identical. This method can be used to detect image manipulation such as cropping, resizing, or color adjustments.

3. ***Feature-based analysis***: This involves identifying unique features within an image and comparing them to other images to determine if they are similar. Features can include edges, corners, and color histograms.

4. ***Metadata analysis***: This involves examining the metadata associated with an image, such as the date and time it was created, the camera used to take the photo, and the location. Differences in metadata can indicate that an image has been altered or manipulated.

It is important to note that no single method is foolproof, and it is often necessary to use a combination of techniques to accurately determine the similarity of digital images. Additionally, the interpretation of the results of a forensic similarity analysis requires expertise in digital forensics and image analysis [3].

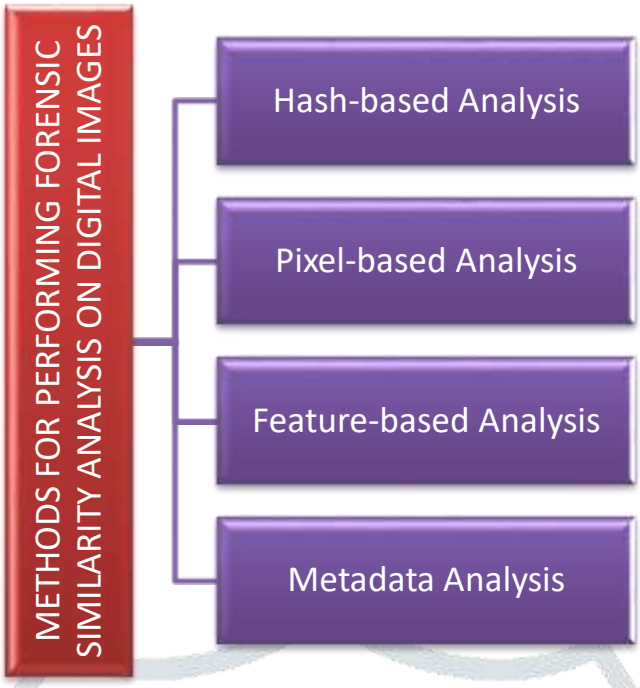

Figure 3. Performing Forensic Similarity Analysis on Digital Images

The technique described in the question is an example of image forensics, which is the field of detecting and analyzing digital image manipulations. In particular, the method is focused on detecting the duplication of scientific images, which is a common problem in scientific research.

The approach is based on training a Convolutional Neural Network (ConvNet) to map images into a 128-dimensional space. The goal of the network is to learn a representation of images that is invariant to common types of image manipulations, such as cropping, resizing, and color correction. This is important because these manipulations can be used to create duplicate images that are difficult to detect with simple image comparison methods [5].

Once the ConvNet is trained, the distance between pairs of images in the 128-dimensional space is computed using the Euclidean distance metric. If two images are duplicates, then their distance will be small (less than or equal to 1 in this case), while if they are different, then their distance will be large (greater than 1). By setting a threshold on the distance, the method can be used to classify pairs of images as either duplicates or non-duplicates.

This approach has several advantages over traditional methods for detecting image duplication, such as using image hashing or feature-based methods. First, the ConvNet is trained to be robust to common types of image manipulations, which makes it more effective at detecting sophisticated manipulations. Second, the method is scalable, since the ConvNet can be trained on large datasets of images. Finally, the method is flexible since it can be adapted to different types of images and image manipulations by retraining the ConvNet on new data [7].

When we apply a convolution operation to a matrix, we are essentially sliding a small window or filter over the matrix and performing a mathematical operation on the values that fall within the window. The filter is usually a small matrix of odd dimensions (e.g. 3x3, 5x5), and it contains numerical values that are used as weights.

The general formula for the convolution operation is:

$$(f * g)[i,j] = \text{sum}(\text{sum}(f[m,n] * g[i-m, j-n])) \qquad (1)$$

where f is the input matrix, g is the filter, and i, j are the indices of the output matrix. The sum is taken over all valid values of m and n.

The convolution operation is used in many image processing applications such as edge detection, blurring, and sharpening. In the context of image forensics, it can also be used to detect image manipulation by comparing the convolution of an original image with the convolution of a potentially manipulated image. If the two convolutions are very similar, then it is likely that the image has not been manipulated. On the other hand, if the convolutions are significantly different, then it is likely that the image has been tampered with [9].

In summary, convolution is a mathematical operation that is used in many image processing applications, including image forensics, and it involves sliding a small window or filter over a matrix and performing a mathematical operation on the values that fall within the window.

The general formula for the convolution operation between a 2D image signal f(x,y) and a 2D kernel g(x,y) can be expressed as:

$$(f * g)(x,y) = \sum\sum f(a,b) * g(x-a, y-b) \qquad (2)$$

where the summation is taken over all possible values of (a,b) such that the kernel g(x,y) is aligned with the image signal f(x,y) at the pixel (x,y).

In other words, to compute the value of the convolution at a given pixel (x,y), we slide the kernel over the image and multiply each pixel of the kernel with the corresponding pixel of the image at a given position. Then, we sum up all these products to obtain the value of the convolution at that pixel.

The result of the convolution operation is a new image signal that represents the response of the original image to the kernel. The properties of the kernel determine the type of image processing operation that is performed by the convolution, such as blurring, sharpening, edge detection, etc.

Image forensics is a field of study that involves the analysis of digital images to determine their authenticity, integrity, and origin. In the context of scientific images and mathematical formulas, image forensics can be used to detect whether an image or formula has been duplicated or manipulated.

One approach to detecting duplication of scientific images is to use digital image analysis techniques such as pixel-level comparisons, histogram analysis, and image feature extraction. These techniques can be used to identify regions of an image that have been copied or pasted, or to detect inconsistencies in lighting, texture, or other visual features that suggest manipulation.

Similarly, mathematical formulas can be checked for duplication by comparing the structure and symbols used in the formula. This can involve analyzing the order of operations, identifying common subexpressions, and checking for consistent use of variables and constants.

Another approach to detecting duplication of scientific images and formulas is to use machine learning techniques, such as deep neural networks. These models can be trained on large datasets of known authentic and manipulated images or formulas, and can learn to identify patterns and features that distinguish between the two.

Ultimately, the effectiveness of any image forensics technique depends on the specific context and the quality of the data being analyzed. However, by combining multiple approaches and using expert judgment, it is often possible to detect and prevent the duplication and manipulation of scientific images and formulas.

There is no one general formula for exposing digital forgeries in scientific images, as the methods used to detect and expose digital forgeries can vary depending on the type of forgery, the image processing techniques used, and the available tools and resources.

However, some common approaches to detecting digital forgeries in scientific images include:

1. ***Image analysis techniques***: Using image analysis software to examine the image for signs of tampering, such as inconsistencies in pixel patterns or lighting, or the presence of duplicate or cloned regions.

2. ***Metadata analysis***: Examining the metadata associated with the image, such as the creation date, author, and camera or device used to capture the image, to look for discrepancies or anomalies that may indicate tampering.

3. ***Comparison to original images***: Comparing the suspect image to the original image or images it purports to depict, looking for inconsistencies or alterations.

4. ***Expert analysis***: Consulting with experts in fields such as image analysis, photography, and forensics to help identify signs of tampering or anomalies in the image.

It is important to note that detecting digital forgeries in scientific images can be a complex and technical process, and may require specialized knowledge, equipment, and software. It is also important to follow established ethical and scientific guidelines for handling and analyzing digital images, to ensure that any conclusions drawn from image analysis are reliable and valid.

The equation provided models the relationship between the true image I0 and the observed image I1. The observed image I1 is the result of applying in-camera processing to the true image, which introduces some noise and other effects. Specifically, the equation can be broken down into several components:

- I0(x,y): the true image at spatial location (x,y)
- gI0(x,y): the in-camera processing applied to the true image at spatial location (x,y)
- K1(x,y): a scaling factor that accounts for any gain or attenuation introduced by the in-camera processing
- N(x,y): the camera noise at spatial location (x,y)

The equation can be read as follows: the observed image I1 at spatial location (x,y) is equal to the true image I0 at that location, with in-camera processing applied (gI0), scaled by K1, and with camera noise N added.

To authenticate an image using the estimated camera noise and extracted image noise, one could compare the statistical properties of the noise in the observed image I1 to those of the estimated camera noise and extracted image noise. If the statistical properties match, then the observed image is likely to be authentic.

Note that this approach assumes that the in-camera processing and camera noise are stationary and do not change over time or with different image conditions. If these assumptions do not hold, the approach may be less reliable.

The equation that describes the relationship between the true image I0(x,y), the in-camera processed image gI0(x,y), the scaling factor K1(x,y), and the camera noise N(x,y) is:

$$gI0(x,y) = K1(x,y) * I0(x,y) + N(x,y) \quad (3)$$

This equation states that the in-camera processing of the true image at spatial location (x,y) results in a scaled image gI0(x,y), where the scaling factor K1(x,y) represents any gain or attenuation introduced by the processing. Additionally, camera noise N(x,y) is also present in the resulting image.

It seems that you have provided information on the size of two classes in a dataset, namely the "forged image class" and the "real image class". The size of each class is as follows:

- Forged image class:
  - Training set: 5111 images
  - Validation set: 2889 images
- Real image class:
  - Training set: 7250 images
  - Validation set: 4259 images

It's important to note that the size of a dataset is just one factor that affects the performance of a machine learning model trained on that dataset. Other factors include the quality of the data (e.g., how representative it is of the problem you're trying to solve), the complexity of the model, the choice of hyperparameters, and so on.

While the size of a dataset is an important factor, it is not the only one that affects the performance of a machine learning model. Other factors that can affect performance include:

1. ***Quality of data***: The quality of data can have a significant impact on the performance of a machine learning model. Low-quality or noisy data can lead to inaccurate or biased results, even with a large dataset.

2. ***Data balance***: The balance of classes in the dataset can also affect the performance of the model. If one class is overrepresented or underrepresented, the model may have difficulty learning to classify that class accurately.

3. ***Model architecture***: The choice of model architecture and hyperparameters can greatly influence the performance of the model. A well-designed model can achieve good results even with a smaller dataset.

4. ***Training methodology***: The training methodology, such as the choice of optimizer, learning rate, and regularization techniques, can also impact the performance of the model.

5. ***Evaluation metrics***: The choice of evaluation metrics can affect the perceived performance of the model. Different metrics may be appropriate for different tasks or datasets.

Therefore, it is important to consider all of these factors when designing and evaluating machine learning models, in addition to the size of the dataset.

function to output the probability of the input image being "fake".
It is important to note that this model is specifically designed for binary classification of real vs. fake images and may not perform well on other image classification tasks. Additionally, while the architecture outlined in the paper has been shown to perform well on certain datasets, it may not be optimal for all applications and may require adjustments or fine-tuning depending on the specific use case, presented in fig [4] & fig [5].

Refer Table [1] & Table [2] The model represents one approach to detecting DeepFake images using a relatively simple CNN architecture. However, it is important to note that the field of DeepFake detection is rapidly evolving, and there may be other approaches or architectures that are more effective for this task.

Table 1. Numerical Results for All Classifiers

Algorithms	Precision	Recall	PPV	NPV	FPR	FNR
SVM	0.916	0.78	0.8178	0.8182	0.204	0.192
ANN	0.9341	0.8214	0.8822	0.8979	0.12	0.1
CNN	0.9687	0.884	0.9221	0.9124	0.16	0.0962

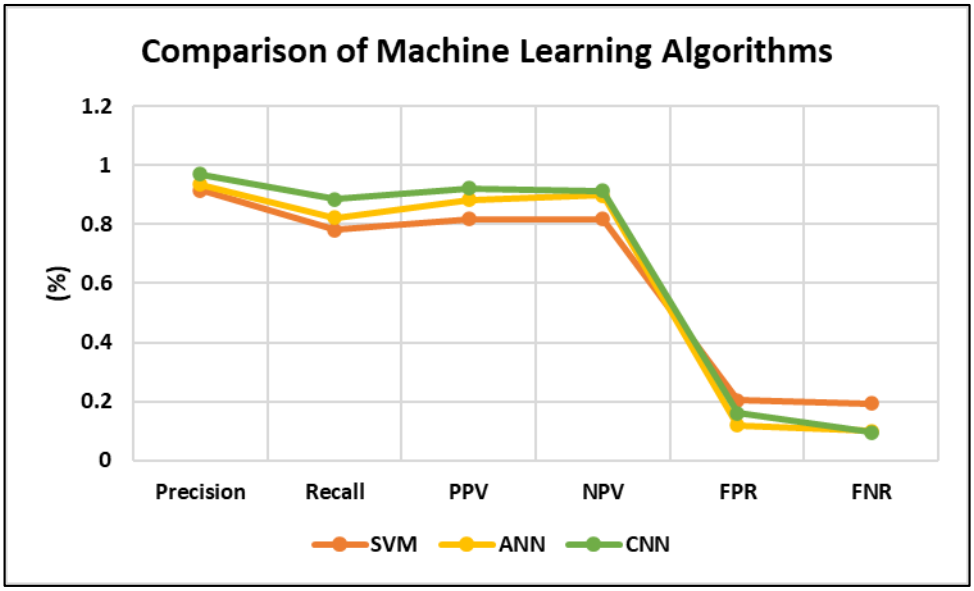

Figure 4. Comparison of Machine Learning Algorithms

Table 2. Comparative Analysis of Classifiers

Performance matrices	SVM	ANN	CNN
Accuracy	0.9222	0.89	0.848
Sensitivity	0.884	0.9	0.78
Specificity	0.976	0.88	0.916
F-measure	0.9224	0.8654	0.8772
G-mean	0.9231	0.8898	0.8441
Precision	0.9687	0.9341	0.916
Recall	0.884	0.8214	0.78

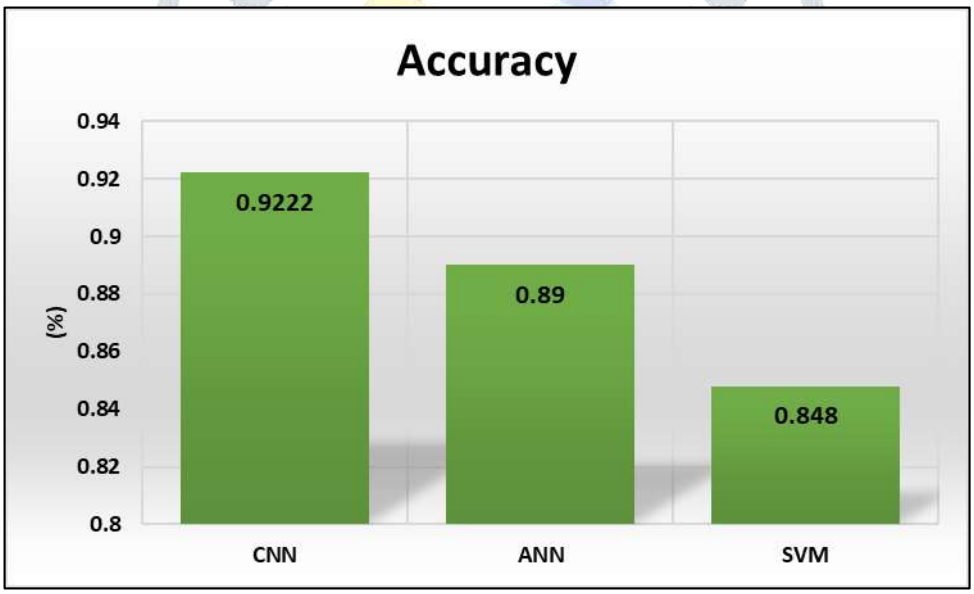

Figure 5. Accuracy

IV. CONCLUSION

The image forgery detection-based CNN algorithm has been discussed in this paper. The copy-move forgery detection is carried out on each image block by certain processing techniques thus reducing the computational difficulty. The performance evaluation displays that the CNN achieves an efficient result in terms of all parameters when compared to the existing algorithms. This process can be further improved by using various algorithms to enhance the accurate classification of forgery image.

REFERENCES

[1] Cao Y, Gao T, Fan L and Yang Q 2012 A robust detection algorithm for copy-move forgery in digital images Forensic Sci. Int. 214 33-43

[2] Kuznetsov A, Myasnikov V 2016 A Copy-Move Detection Algorithm Using Binary Gradient Contours International Conference on Image Analysis and Recognition, ICIAR 9730 349-357

[3] Bayar B, Stamm M C 2017 On the robustness of constrained convolutional neural networks to jpeg post-compression for image resampling detection Proceedings of the 42nd IEEE International Conference on Acoustics, Speech and Signal Processing

[4] Rao Y, Ni J 2016 A deep learning approach to detection of splicing and copy-move forgeries in images IEEE International Workshop on Information Forensics and Security (WIFS) 1-6

[5] Amerini I, Uricchio T, Ballan L, Caldelli R 2017 Localization of JPEG double compression through multi-domain convolutional neural networks IEEE Conference on Computer Vision and Pattern Recognition Workshops 1865-1871

[6] Simonyan K, Zisserman A 2014 Very deep convolutional networks for large-scale image recognition arXiv preprint arXiv:1409.1556

[7] Sutthiwan P, Shi Y Q, Zhao H, Ng T-T and Su W 2011 Markovian rake transform for digital image tampering detection Transactions on data hiding and multimedia security VI 1-17

[8] Wang W, Dong J and Tan T 2009 Effective image splicing detection based on image chroma ICIP. IEEE 1257-1260

[9] Wang W, Dong J and Tan T 2010 Image tampering detection based on stationary distribution of markov chain ICIP. IEEE 2101-2104

[10] Lin Z, He J, Tang X and Tang C-K 2009 Fast, automatic and finegrained tampered jpeg image detection via DCT coefficient analysis Pattern Recognition 42(11) 2492-2501