e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:07/July-2021 Impact Factor- 5.354 www.irjmets.com

DETECTION OF REAL AND FAKE IMAGE USING DEEP LEARNING ALGORITHM

Dr Chanda V Reddy*1, Anusha H*2, Dhanush N*3, Madhushree T P*4, Nischay P*5

*1Professor And Head , Department Of Telecommunication, K S Institute Of Technology, VTU, Banglore, Karnataka, India.

*2,3,4,5Student, VTU, Telecommunication, K S Institute Of Technology, Banglore, Karnataka, India.

ABSTRACT

In the present generation, social media is a big advantage for an individual to grow. On the other hand, we can't neglect the fact that the it's a huge platform for negativity too. With the rapid progress of recent years, techniques that generate and manipulate multimedia content can now provide a very advanced level of realism. The boundary between real and synthetic media has become very thin. On the one hand, this opens the door to a series of exciting applications in different fields such as creative arts, advertising, film production, video games. On the other hand, it poses enormous security threats. Software packages freely available on the web allow any individual, without special skills, to create very realistic fake images and videos. These techniques can be used to manipulate public opinion regarding anything and create chaos. In this paper, we would like to overview few major facts and figures regarding exceeding image forgery techniques that exists and propose a better way on how to detect these forgeries and fakes.

Keywords: GAN, ELA, Deep Learning, Convolutional Neural Networks, Fake Colorized Image Detection.

I. INTRODUCTION

Fake images have become a central problem in the last few years, especially after the advent of the so-called deep fakes, i.e., fake images manipulated with the help of powerful and easy-to-use deep learning tools, like auto encoders (AE) or generative adversarial networks (GAN). With this technology, creating realistic manipulated media assets may be very easy, provided one can access large amounts of data. The very same technology, however, can also be used for malicious purposes, like creating fake porn images to blackmail people, or building fake-news campaigns to manipulate the public opinion.

These editing of images can be easily done with taking care that it is not so obvious to tell that it is a fake image by using powerful editing apps like adobe photoshop or GIMP etc. Using such conventional editing app methods, images can be easily modified, obtaining realistic results that can fool even a careful observer. An example of skillfully manipulated images that have been disseminated on the Internet in recent years to spread false news are shown in figure below.

Now a days the trust on digital image is less but now in order the gain the trust of the digital image there are few techniques to detect the tampering of the image. The first method can be explained as the Watermark Method i.e., Digital watermarking has been proposed as a means by which an image can be authenticated [1]. The drawback of this approach is that a watermark must be inserted at the time of recording, which means that every camera doesn't come equipped with watermark. So, in order to overcome the disadvantage Pixel based method is useful and that can be explained as mentioned below. Pixel Based in these there are three methods to detect tampering which maybe a direct method or an indirect method and they (1) Cloning (2) Resampling (3) Splicing. The Cloning method is image manipulations is to clone (copy and paste) portions of the image to conceal a person or object in the scene. When this is done with care, it can be difficult to detect cloning visually. And since the cloned regions can be of any shape and location, it is computationally impossible to search all possible image locations and sizes. Two computationally efficient algorithms have been developed to detect cloned image regions. So according to the first author first apply a block discrete cosine transform (DCT). Duplicated regions are detected by lexicographically sorting the DCT block coefficients and grouping similar blocks with the same spatial offset in the image then according to the second author apply a principal component analysis (PCA) on small fixed size image blocks to yield a reduced-dimension representation.

According to the detection of tampering of image this watermark method is useful to detect the medical information security like integrity, authenticity [2]. So, in this watermark method there is an secret key between the transmitter and the receiver . So, whenever the information is exchanged between them then they

need to press the secret key so that the proper watermark is showed on the paper so that no tampering can be easily done without any difficulty. And also there is an reverse watermark process which is advantage of reversible watermarking over other competitors is that it leaves the field free for any desired image processing.

Due to the advancement of image tampering it is difficult to distinguish the original image from an altered image and this can be solved by the correlation between the bit planes as well the binary texture characteristics within the bit planes will differ between an original and a doctored image [3]. This is actually done by assuming that altering an image changes the correlation between and within bit planes. Therefore, the quantal-spatial correlation between the bit planes of the original image will differ from that of the bit planes of the doctored images. Once that is done there are several features extracted from the image measures are based on the bit-by-bit matching between the corresponding pixel positions of the two images by using that algorithm known as Sequential Floating Forward Search (SFFS) algorithm to select the best features and Linear Regression Classifier for classification.

II. METHODOLOGY

(A) Back end of the project:

In Back-end of the project the python code that will be running is the Error Level Analysis and Convolution Neural Network to detect the given Image is Real or a Fake image . And Each topic is explained below with details.

(1) Error Level Analysis:

Error Level Analysis is a forensic method to identify portions of an image with a different level of compression. The technique could be used to determine if a picture has been digitally modified. They result in poor quality compressed images.

Error Level Analysis (ELA) permits identifying areas within an image that are at different compression levels. With JPEG images, the entire picture should be at roughly the same level. If a section of the image is at a significantly different error level, then it likely indicates a digital modification.

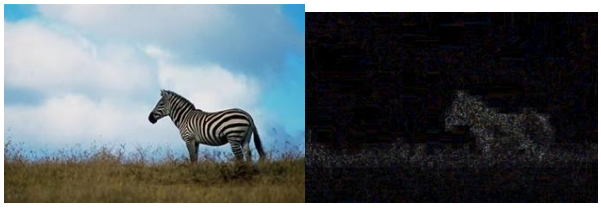

Fig1: ELA with Real Image

Fig 2: ELA with Fake Image

(2) Convolutional Neural Network (CNN):

In deep learning, a convolutional neural network is a class of deep neural networks, most commonly applied to analysing visual imagery.

A convolutional neural network consists of an input and an output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of a series of convolutional layers that convolve with a multiplication or other dot product.

(3) Layers used in this proposed model:

1. Convolution2D
2. Maxpooling2D
3. Flatten
4. Dropout

5. Dense

Requirements needed here are:

- Hardware requirements
  - System processor: i7
  - Hard Disk: 500GB
  - Ram: 8GB / 12GB
  - Android device
- Software requirements
  - Operating system: windows 8 / 10 (64 bits OS)
  - Programming language: Python 3
  - Framework: Anaconda
  - Libraries: Keras, Tensor flow
  - IDE: Jupyter Notebook

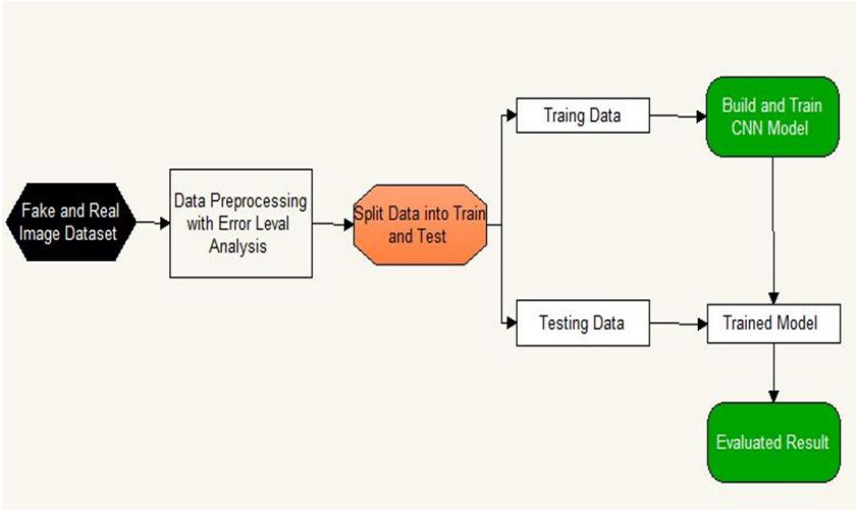

Fig3: Block diagram

The proposed system followed by this Architecture diagram. First, we should collect the dataset then we pre-process the data with Error Level Analysis (ELA). Once we done that, we should split the data into train and test the training data will be feed through inside the Convolutional Neural Network (CNN)for train the model. Once it's getting trained, we use that trained model for evaluate the result for our test data.

(B) Front-End of the project:

In this Process an app is developed particularly for android using android development studio. The only reason why is the app is being developed for android is it allows the installation from the unknown whereas iOS, windows and other platform doesn't allow. The app is created using Android Development Studio because it is easy to create an app and there is no prior coding language to be learnt since the app itself can write the code when the users design the app by using layout, buttons & drawable. Once the app is been created it should be synched with an online cloud-based platform to connect with the python code. To do that there is an online cloud platform known as Firebase. .The Firebase is a Realtime Database and a cloud-hosted database in which data is stored as JSON format. The data is synchronized in real-time to every Front-end connected app. And the Firebase Realtime Database lets your secure access to the database directly from Front-end code. Firebase allows syncing real-time data across all devices - iOS, Android, windows, Linus etc. So once the FireBase is created, we need to copy the package code given by the FireBase and then paste it in the app developed by android to sync them online so that the image can be copied online and then the firebase can transfer it to the back-end of the code and process the image and then transfer back the image as the results to the Front-end of an app. So with the help of Firebase, we can sync the Frontend and back-end of an app.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
Volume:03/Issue:07/July-2021 Impact Factor- 5.354 www.irjmets.com

III. MODELING AND ANALYSIS

The proposed system followed by this Architecture diagram. First, we should collect the dataset then we pre-process the data with Error Level Analysis (ELA). Once we done that, we should split the data into train and test the training data will be feed through inside the Convolutional Neural Network (CNN)for train the model. Once it's getting trained, we use that trained model for evaluate the result for our test data.

I. RESULTS AND DISCUSSION

As proceeding to the final stage of project by performing the methods mentioned above, we can obtain the results of detection of images which are being real or Fake. In initial Page it asks for valid Username and password to gain access to the working of app. Then once the image is selected from the gallery of the phone it starts uploading the image to the cloud-based platform called Firebase and links to the back end of the project and run the algorithm and produces the accurate result of whether the image is real or fake.

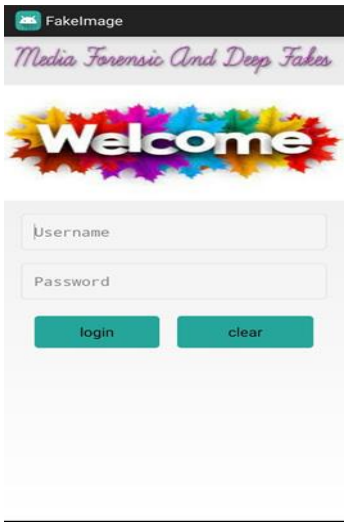

Fig 4: Generated app asking for login credentials.

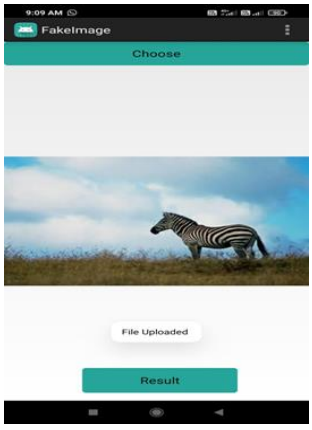

Fig 5: Image uploaded on the app.

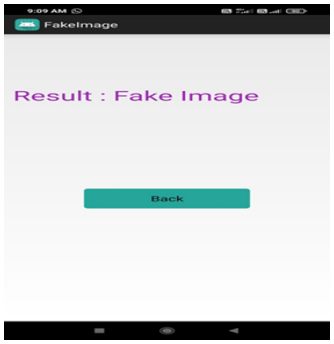

Fig 6: Result produced on the app.

Here in our project, any real time image can be uploaded in this app and identified whether the image is real or fake.

IV. CONCLUSION

In the backend of the project, we have built the deep learning algorithm that is CNN with ELA as data preprocessing stage where we used bifurcated real and fake images as data set to train and validate the model. We saved this model in json format to make it easy for the further interface. And android app was built using android studio code and was connected to previously built model through Firebase. This app acts like a platform where we can upload real time image to determine whether the uploaded image is real or fake.

V. REFERENCES

[1] H. Farid, "Image forgery detection," IEEE Signal Processing Magazine, Vol. 26, no. 2, pp. 16–25, 2009

[2] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation de taction with binary similarity measures," in Proc. European Signal Processing Conf., Turkey, 2005

[3] S. Bayram, I. Avcibas, B. Sankur, and N. Memon, "Image manipulation detection," J. Electron. Imaging, vol. 15, no. 4, p. 41102, 2006.

[4] Alessandro. Piva, "An overview on image forensics," ISRN Signal Processing, pp. 1–22, 2012.

[5] G. W. Meyer, H. E. Rushmeier, M. F. Cohen, D. P. Greenberg, and K. E. Torrance, "An experimental evaluation of computer graphics imagery," ACM Transactions on Graphics, vol. 5, no. 1, pp. 30–50, 1986.

[6] T. de Carvalho, C. Riess, E. Angelopoulou, H. Pedrini, and A. Rocha, "Exposing digital image forgeries by illumination color classification," IEEE Trans. Inf. Forensics Security, vol. 8, no. 7, pp. 1182–1194, 2013.

[7] ATHERTON, P., AND CAPOREAL, L. A subjective judgement study of polygon based curved surface imagery. CHI'85 Conference on Human Factors in Computing Systems (San Francisco, Calif., Apr. 14-18). ACM/SIGCHI, New York, 1985.

[8] A. Piva, "An Overview on Image Forensics," ISRN Signal Processing, vol. 2013.

[9] Yi-Lei Chen and Chiou-Ting Hsu, "Detecting Doubly Compressed Images Based on Quantization Noise Model and Image Restoration", 2009 IEEE International Workshop on Multimedia Signal Processing, 23 October 2009, 978-1-4244-4652-0/09.