Juni Khyat
(UGC Care Group I Listed Journal)

ISSN: 2278-4632
Vol-10 Issue-5 No. 14 May 2020

IMAGE FORGERY DETECTION

P. Lavanya[1] B. Jagruthi[2] M. Srinidhi[2] P.Mallesh[2]
[1]Assistant Professor, Department of CSE,[2]Final year B.Tech. Students, Department of CSE,
Sreyas Institute of Engineering and Technology, Hyderabad, Telangana, India

ABSTRACT

Many fake images are spreading through digital media now a days. Detection of such fake images is inevitable for the unveiling of the image based cybercrimes. Forging images and identifying such images are promising research areas in this digital era. The tampered images are a detected using neural network which also recognizes the regions of the image that have been manipulated and reveals the segments of the original image. It can be implemented on Desktop Application on Java platform and hence made available to common users. The compression ratio of the foreign content in a fake image is different from that of the original image and is detected using Error Level Analysis. Another feature used along with compression ratio is image metadata. Although it is possible to alter metadata content making it unreliable on its own, here it is used as a supporting parameter for error level analysis decision.

1.Introduction

In this technological era a huge number of people have become victims of image forgery. A lot of people use technology to manipulate images and use it as evidences to mislead the court. So to put an end to this, all the images that are shared through social media should be categorized as real or fake accurately. Social media is a great platform to socialize, share and spread knowledge but if caution is not exercised, it can mislead people and even cause havoc due to unintentional false propaganda. While manipulation of most of the photoshoped images is clearly evident due to pixelization & shoddy jobs by novices, some of them indeed appear genuine. Especially in the political arena, manipulated images can make or break a politician's credibility. Current forensic techniques require an expert to analyze the credibility of an image. We implemented a system that can determine whether an image is fake or not with the help of machine learning and thereby making it available for the common public. This paper will unfold into three sections whereby first will focus on the second will focus on the Implementation details while the last part showcase the experimental result.

1.1 Motivation

This project provides two level analysis for the image. At first level, it checks the image metadata. Image metadata is not that much reliable since it can be altered using simple programs. But most of the images we come across will have non-altered metadata which helps to identify the alterations. For example, if an image is edited with Adobe Photoshop, the metadata will contain even the version of the Adobe Photoshop used.

1.2 Problem Definition

Since the invention of photography, individuals and organizations have often sought ways to manipulate and modify images in order to deceive the viewer. Existing projects have worked on the comparison of image forgery detection methods, these are often limited in scope and only compare variants of the same algorithm on images that are specifically created for that type of method. There are also forged images which cannot be detected by the existing applications.

1.3 Objective

The objective of this project is to identify fake images (Fake images are the images that are digitally altered images). We approached the problem using machine learning and neural network to detect almost all kinds of tampering on images.

2. Existing System

The problem with existing fake image detection system is that they can be used detect only specific tampering methods like splicing, coloring etc. Using latest image editing software's, it is possible to make alterations on image which are too difficult for human eye to detect. Even with a complex neural network, it is not possible to determine whether an image is fake or not without identifying a common factor across almost all fake images.

2.1. Disadvantages of Existing system

- Unable to differentiate the copy paste regions.
- Not robust against compression varient.
- It cannot be used for color images.
- Forged regions with JPEG compression cannot be detected.

- Computational complexity is less.

3.Proposed System

This project provides two level analyses for the image. At first level, it checks the image metadata. Image metadata is not that much reliable since it can be altered using simple programs. But most of the images we come across will have non-altered metadata which helps to identify the alterations. For example, if an image is edited with Adobe Photoshop, the metadata will contain even the version of the Adobe Photoshop used.

In the second level, the image is converted into error level analyzed format and will be resized to 100px x 100px image. Then these 10,000 pixels with RGB values (30,000 inputs) is given in to the input layer of Multilayer perception network. Output layer contain two neurons. One for fake image and one for real image. Depending upon the value of these neuron outputs along with metadata analyzer output, we determine whether the image is fake or not and how much chance is there for the given image to be tampered.

3.1.System Architecture

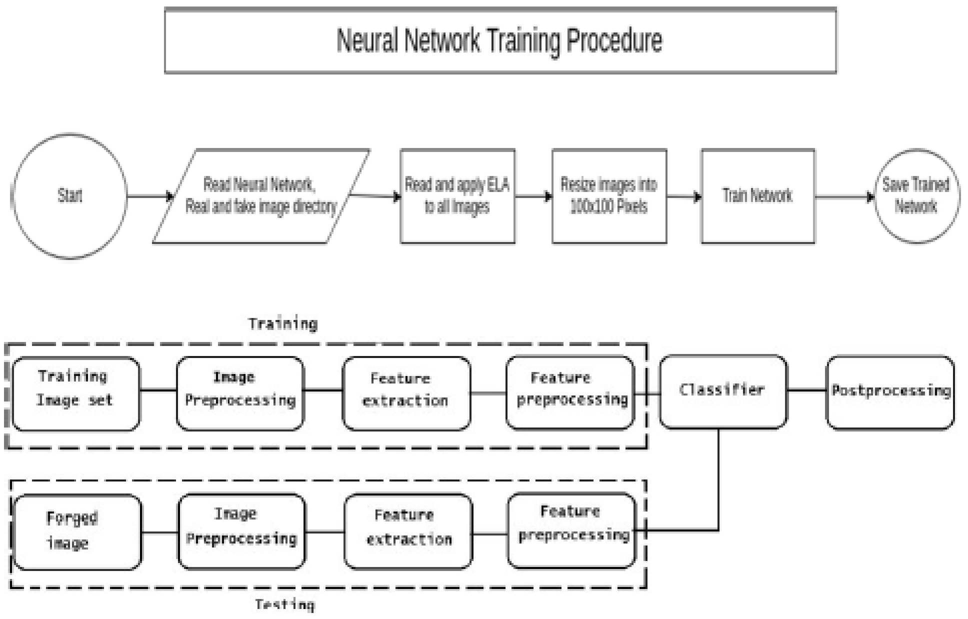


Fig1: Architecture of Image Forgery Detection

4.Implementation & RESULTS

Project implementation (or project execution) is the phase where visions and plans become reality. It is the stage of the project when the theoretical design is turned out into a working system. The implementation stage involves careful planning, investigation of the existing system and it's constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods. It is important to take into account that independently of the nature of the project, implementation takes time, usually more than it is planned, and that many external constraints can appear, which should be considered when initiating the implementation step.

4.1.Output Screen

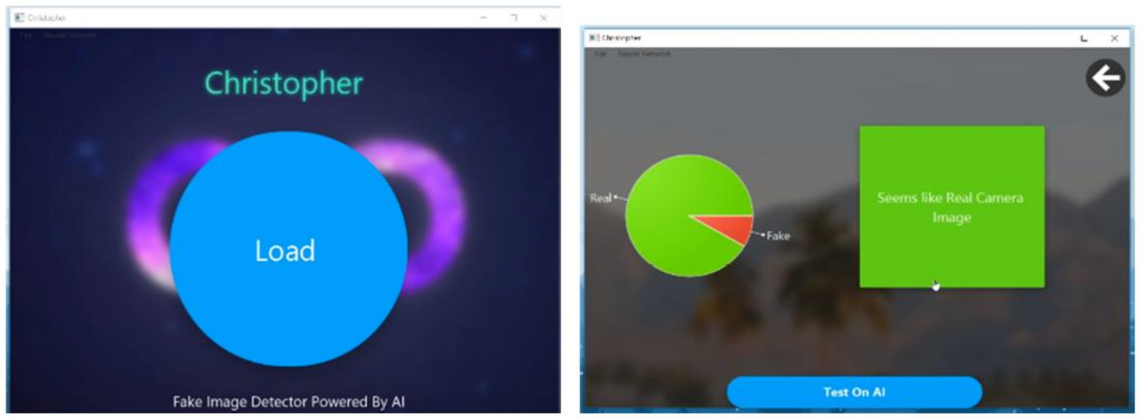
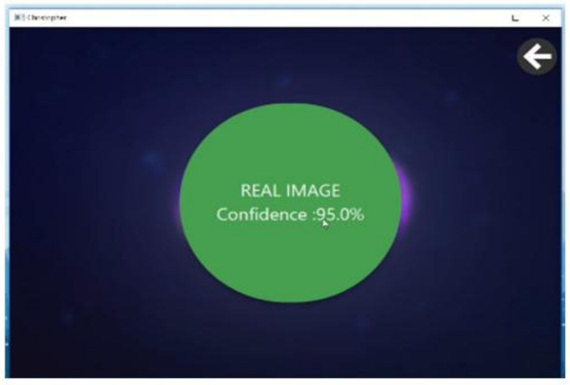

Christopher
Load
Fake Image Detector Powered By AI
Real
Fake
Seems like Real Camera Image
Test On AI

Fig.2 Output Screen for the user to load an image.



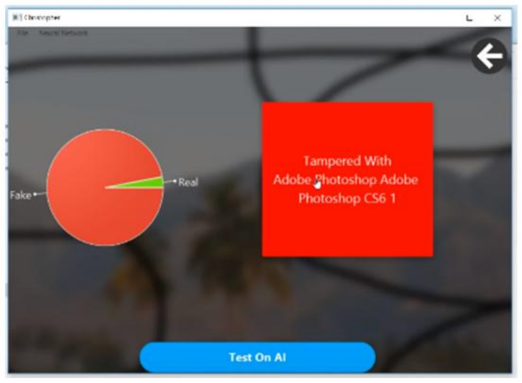


Fig. 3 Output Screen displaying the result generated

Fig.4 Output Screen displaying the result message.

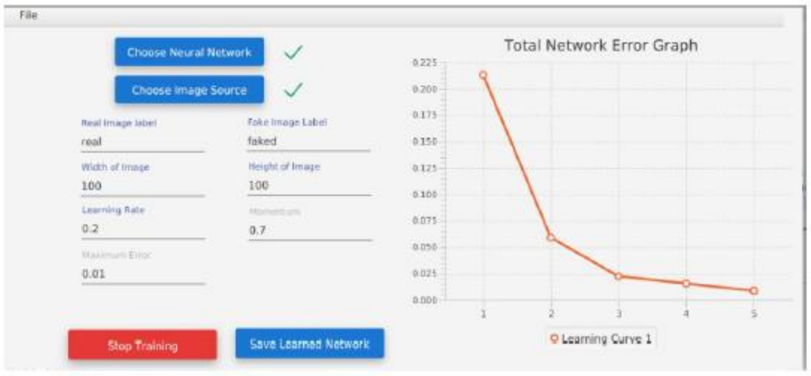
4.2.Design of Test Cases

Tested	Test name	Inputs	Expected output	Actual Output	Result
1	Splitting data into classes	data	input taken	successfully splitted	pass
2	load dataset	dataset	dataset loaded	successfully loaded	pass
3	Splitting dataset into training and testing	dataset	splitted to training and testing	successfully splitted to train data and test data	pass
4	load cnn classifier	sklearn module	model created	successfully model created	pass
5	training set evaluation	train data	training done	successfully trained	pass
6	Validating test dataset	test data	test dataset validated	successfully validated	pass
7	getting accuracy	test dataset	accuracy in percentage	successfully got accuracy	pass
8	load test image	test image	detecting harmful or not	successfully detected	pass
9	print result	test image	printing malignant or begant	successfully printed	pass

ISSN: 2278-4632
Vol-10 Issue-5 No. 14 May 2020

4.3.Experimental Result

Metadata analysis has shown promising result in non-shared images. It is able to detect anomaly in all ‘photoshopped’ or ‘gimped’ images under a very small processing. It failed on images shared through WhatsApp, Google+ etc. Moreover, it became completely erroneous when images with manipulated metadata given. Neural network is trained with CASIA dataset. The dataset contains 7491 real images and 5123 tampered images under varying sizes. All the images are preprocessed to 100x100 pixels so that total pixel values to be fed into the neural network will be 30,000. From the dataset we have used 4000 real and fake images for training. Remaining images were used for testing of the neural network. Table 3 shows various neural network configurations and corresponding neural network efficiency. Best is achieved when learning rate set to 0.2 and momentum to 0.7.



Real Image label: real	Fake Image Label: faked
Width of Image: 100	Height of Image: 100
Learning Rate: 0.2	Momentum: 0.7
Maximum Error: 0.01	

5 Conclusion

Neural network has been successfully trained using the error level analysis with 4000 fake and 4000 real images. The trained neural network was able to recognize the image as fake or real at a maximum success rate of 83%. The use of this application in mobile platforms will greatly reduce the spreading of fake images through social media. This project can also be used as a false proof technique in digital authentication, court evidence evaluation etc. By combining the results of metadata analysis (40%) and neural network output (60%) a reliable fake image detection program is developed and tested.

REFERENCES

[1] A picture’s worth, Digital Image Analysis and Forensics, N Krawetz - 2007 Ph D, Hacker Factor Solutions

[2] http://imagej.net/Welcome ImageJ is an open source image processing program designed for scientific multidimensional images.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] http://forensics.idealtest.org/ CASIA v2.0 CASIA V2.0 is with larger size and with more realistic and challenged fake images by using post-processing of tampered regions. It contains 7491 authentic and 5123 tampered color images.

[4] http://neuroph.sourceforge.net/ Neuroph Framework Neuroph is lightweight Java neural network framework to develop common neural network architectures. Itcontains well designed, open source Java library with small number of basic classes which correspond to basic NN concepts.

[5] https://github.com/drewnoakes/metadata-extractor Metadata-extractor is a straightforward Java library for reading metadata from image files. [6] https://www.github.com/afsalashyana/FakeImageDetection GitHub repositor for fake image detector desktop application written in javafx.