Analysis of image forgery detection using convolutional neural network

Chiluveru Gnaneshwar,
Manish Kumar Singh and
Satyendra Singh Yadav*

Department of Electronics and Communication Engineering,
NIT Meghalaya,
Shillong, 793003, India
Email: b16ec024@nitm.ac.in
Email: b16ec028@nitm.ac.in
Email: satyendra@nitm.ac.in
*Corresponding author

Bunil Kumar Balabantaray

Department of Computer Science and Engineering,
NIT Meghalaya,
Shillong, 793003, India
Email: bunil@nitm.ac.in

Abstract: Prior to the age of cameras, if someone wanted to see/verify any incident or document, then one must go to that place and verify. The fact is that no one ever questions once someone has verified something with their own eyes. Nowadays, with the rapid development of new technologies, one cannot be sure of an image, which one is a copy of the sight or not a sight itself. Such types of verifications are not possible in the current time due to the development of varieties of advanced image editing tools like Corel draw, Photoshop, GIMP, etc. These are low cost and open-source tools for the users and frequently used to make memes on social media websites. This paper presents an image forgery detection using convolutional neural networks (CNNs/ConvNet). The error level analysis (ELA) method is discussed in detail for image forgery detection. The binary decision of CNN-based model helps in declaration of an image aptness for official uses. The CNN model has been trained for the Kaggle dataset and detailed simulations have been carried out to validate the accuracy and precision of the proposed model.

Keywords: image forgery detection; convolutional neural network; CNN; error level analysis; ELA; machine learning; ML; deep learning; DL.

Reference to this paper should be made as follows: Gnaneshwar, C., Singh, M.K., Yadav, S.S. and Balabantaray, B.K. (2022) 'Analysis of image forgery detection using convolutional neural network', *Int. J. Applied Systemic Studies*, Vol. 9, No. 3, pp.240–260.

Biographical notes: Chiluveru Gnaneshwar has completed his BTech in the Department of ECE National Institute of Technology, Meghalaya, India. He has completed his final year project under the supervision of Dr. S.S. Yadav. His research interest is machine learning and data science.

Manish Kumar Singh has completed his BTech in the Department of ECE National Institute of Technology, Meghalaya, India. He has completed his final year project under the supervision of Dr. S.S. Yadav. His research interest is machine learning and data science.

Satyendra Singh Yadav has received his BEng in ECE from the RGPV Bhopal, India, in 2012. In 2018, he received his PhD from the NIT, Rourkela, India. He was a part of the INESC-ID, Instituto Superior Técnico Lisbon, Portugal, under India-EU NAMASTE Mobility Project during 2015 to 2016. He is currently working as an Assistant Professor in the Department of ECE at NIT Meghalaya, India. His research interests include wireless communication, resource allocation, parallel computing, machine learning, as well as GPU acceleration for 5G and beyond wireless systems. Since 2014, he has been a member of the IEEE.

Bunil Kumar Balabantaray received his BTech in Information Technology and MTech in Computer Science and Engineering from BPUT, India, in 2005 and 2010, respectively. He completed his PhD from the National Institute of Technology Rourkela, India, in 2017. Currently, he is working as an Assistant Professor in the Department of Computer Science and Engineering, National Institute of Technology, Meghalaya. He has 15 years of teaching and research experience. He is serving as a reviewer of many journals of national and international repute. His area of research includes IoT, cloud computing, computer vision, robotics and biomedical image processing.

1 Introduction

Images are a large source of information for its viewer. In the current times of social media and the internet, digital images have certainly proved its powerfulness and convenience. Requirements of images are endless in medical imaging, digital forensics, intelligence, journalism, photography and court of law as evidence (Selamat et al., 2008). Therefore, one of the important and common uses is as the proof of incident of the past activities. Since it enables one to remember a thousand words in just one image, in the current decade, the reliability of these images has been a topic of discussion (Selamat et al., 2008; Shwetha and Sathyanarayana, 2016). As the technologies are outgrowing to the human curiosity, editing of an image is not a fresh thought as per one's requirement. Availability of contemporary technologies in the form of software with easy user interface, such as Photoshop, GIMP, Premiere, Vega, Corel draw, etc. make its use even more widespread (Fineman, 2012). The thought of editing the image tampers the originality of an image and such practice is called image forgery (Bharti and Tandel, 2016). Image forgery generally is any manipulation of the digital image to hide some meaningful and significant data in the image. The availability of such fine software stated above makes it difficult to find out the manipulated region.

Based on the approach associated with the method, there are following types of image forgery detection techniques (IFDTs):

1 *Active approach:* The active method inserts certain authentic details (watermark) or signatures with the photo at the time of capturing or after capturing the photo. It is then modified by employing digital watermarking technique or digital signature

technique or both (Shwetha and Sathyanarayana, 2016) and thereafter it is shared to others.

2 *Passive approach:* The passive method does not insert any details for validation motive. It functions totally by studying the binary data of digital images. Such images in which no watermarking and signatures are there to cross-verify are called a blind image. Researchers divided passive approach in different ways based on editing techniques (Birajdar and Mankar, 2013).

The passive tools can be nearly classified into five classes (Farid, 2009):

1 *Pixel-level:* Procedure distinguishes measurable inconsistencies presented at the pixel level.

2 *Format-level:* Procedures influence the measurable relationships presented by a lossy pressure scheme.

3 *Camera level:* Procedures abuse ancient rarities presented by the camera focal point, sensor, or on-chip post-handling.

4 *Physical methods:* Explicitly determine and identify peculiarities in the three-dimensional cooperation amongst physical items, light and the camera.

5 *Geometrical methods:* Make estimations of articles on the planet and their positions.

Passive image forensic tools are also popularly classified in three categories, as follows (Birajdar and Mankar, 2013):

a copy move image forgery

b image splicing

c image retouching.

Above techniques with involvement of machine learning (ML) detects different types of forgery with different accuracies (Doegar et al., 2019; Guo et al., 2016). The discussion of the pre-existing techniques or approaches to detect image forgery is very important to evolve a new detection technique of our own. Convolutional neural network (CNN/ConvNet) is a deep learning (DL) rooted algorithm that takes images from input and assigns weights to various components of the image and uniquely identifies different images. The processing involved in ConvNet is much lower than processing in other competing algorithms. The different complex algorithms are widely targeted on massively parallel architectures (Lopes et al., 2019; Yadav et al., 2019). Error level analysis (ELA) identifies the area of forgery with difference in compression levels. For an image in joint photographic experts' group (JPEG) format, the compression level should be approximately at the same level. Digital forgery is detected when there is a significant difference in the error level.

Following the introduction, the rest of the paper is organised as Section 2 provides the detailed state-of-the-art and motivation of this work. Section 3 discusses the ConNet architecture and some important terminologies related to ConvNet. Section 4 focuses on the ELA of image. Proposed model of image forgery detection along with results and analysis is discussed under Section 5. Finally, Section 6 provides the concluding remarks on this work.

2 Related work

An IFDT for any type of forgeries has the following motives:

1 to detect forged regions with accuracy

2 to improve the accuracy

3 to regionalise the forgery

4 time of detection must be as low as possible.

IFDT are categorised in two different types, as follows:

- image processing
- DL.

Here, Ansari et al. (2014) and Hacker Factor (http://www.hackerfactor.com/) used ELA and Gu et al. (2018), Liu and Deng (2015), Han et al. (2019) and Bayar and Stamm (2016) used CNN for image forgery detection which are integral part of image processing and DL, respectively. Bharti and Tandel (2016) surveys different techniques for approaching the detection of image forgeries. Farid (2009) categorises image forgeries in different passive approaches which were of great importance for the development of initial knowledge and learning of IFDT. Dureja and Pahwa (2018) survey the techniques on how to recover a forged image based on different retrieval techniques. Krizhevsky et al. (2012) discusses the use of CNN to classify different images categories they obtained from the ImageNet database. Mishra et al. (2013) discusses speed up robust features (SURF) and hierarchical agglomerative clustering (HAC) techniques for image forgery specifically for images which use region duplication. Mushtaq and Mir (2014) detects forgery based on statistical features such as average entropy, entropy, skewness, and kurtosis of a forged image. Popescu and Farid (2005) uses the image capturing technique used by cameras, employing a sensor in combination with a colour filter array (CFA) to detect a forgery in images. Uliyan et al. (2016) discusses the detection of a forged blurred area with a new system which combines blur metric evaluation and phase congruency. Walia and Kumar (2019) survey systematically the problems, questions, and existing techniques in the field of digital image forgery. Zhang et al. (2016a) discusses the detection of image forgery using CNN. Zhang et al. (2016b) discusses the detection of regions of image forged with the help of neural networks and DL. Zhou et al. (2017) discusses a new approach of block-based CNN for image forgery detection which consists of regions with convolutional neural networks (RCNN) for processing of each block.

Doegar et al. (2019) have used the AlexNet model of object detection to detect image forgery. Bharti and Tandel (2016) surveys various techniques by which forged images are detected. The article comes with a useful conclusion with comparison of some parameters, merits and demerits. Online resources in Fotoforensics (http://fotoforensics.com/tutorial-ela.php) have shown great impact in the ELA knowledge and development. Chen et al. (2015) discusses CNN for image forensics using median filtering, a nonlinear digital filtering technique. Ansari et al. (2014) reviews the passive image forgery detection based on pixels. In this article, various ideas are given on pixel-based forgery detection. Krawetz et al. (Hacker Factor, http://www.hackerfactor.com/) elaborates ELA

and its use in image forensics and Birajdar and Mankar (2013) surveys a wide range of passive techniques used to detect image forgeries. The latest advancement of CNNs in IFD, its use and analysis has been discussed by Gu et al. (2018). Liu and Deng (2015) show how to complete CNN training effectively with a small data size for pre-processing of models. Han et al. (2019) provides a discussion for CNN on any generic data and discusses the difference in approaches for an image and other data. Bayar and Stamm (2016) presented the detection of image forgeries with a new convolutional layer as a part of DL. The motivation to this work can be listed out in brief as follows:

- The motivation behind this work is fake news. Fake news is a news which is not true and is only generated to do social damage to any society, agency, person, or an entity. In this world of social media posting anything and it is getting shared by thousands of people is not an uncommon thing to happen. While fake news in the form of a forged image or edited video or with wrong subtitle can totally influence many unaware and innocent people to think and can result in social losses to societies.
- The use of image forgery in any crime scene is quite popular and has significant importance. The use of a reliable IFDT can help media houses to cover the news for large geographical areas and verify fake news as part of the process. The work is also interesting because of its relationship with image modification and its study using DL. So, it gradually becomes an interesting research field to get involved in.

The use of ML/DL technique was an obvious choice for this work by considering its popularity, fastness, accuracy and reliability. The repetitive passing of new data in a ML model makes it independently adaptable. By training a model with large datasets the reliability of a model increases. ML comes under a big umbrella of artificial intelligence (AI) which makes it even more future oriented, as the field is mostly stated as the future of all technologies.

3 CNN basics

CNNs/ConvNet consist of three different words literally which describe the functioning of the DL-based model. Let us understand the three words as follows:

a *Convolution:* It is the operation between two functions, and it produces a third function which describes the relation between the two original functions. Here, one of the original functions is input data and the other is the parameters which change with each training data to get the desired output. A convolution is basically sliding a filter over the input image data.

b *Neuron:* In ML, a neuron is the basic building block of a model, it calculates a weighted average of its inputs with necessary biases and at last adds nonlinearity to the result.

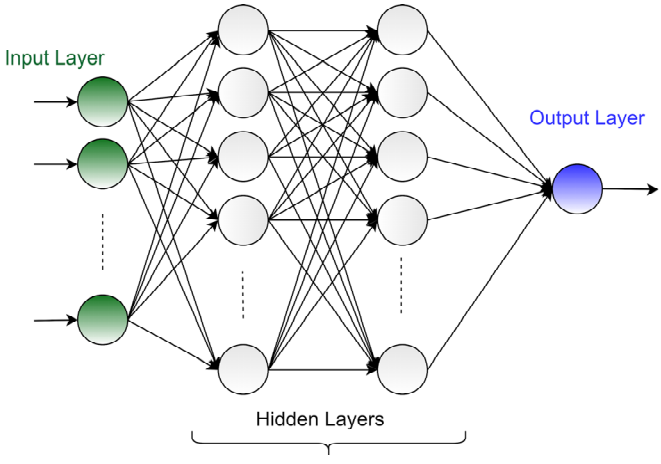
c *Network:* In the world of data communication, we pass data through many layers of routers, access points, or hubs thus forming a network. Hence, here we pass input data and its modifications through many layers thus the model is called a network.

The CNN is a convolution-based neural network which comprises learnable weights and biases to complete a task. CNN accepts an image's data as input (Das et al., 2020; Nayak et al., 2020). Taking an image directly as input saves several variables and confusions in the model. The model does dot product between inputs and filter matrix basically of small size sliding all over the image. The output at the end of the network however is given with a percentage value for each possibility.

3.1 Basic neural networks

Basic neural network (BNN) takes a single vector input. It consists of many hidden layers like CNN, these layers further consist of many neurons. Layer wise study tells that every neuron of the next layer is connected to the neurons of the previous layer. At the end, a fully connected (FC) layer is attached to all previous layer neurons and produces an output in the form of percentage values. A BNN is presented in Figure 1.

Figure 1 A BNN (see online version for colours)



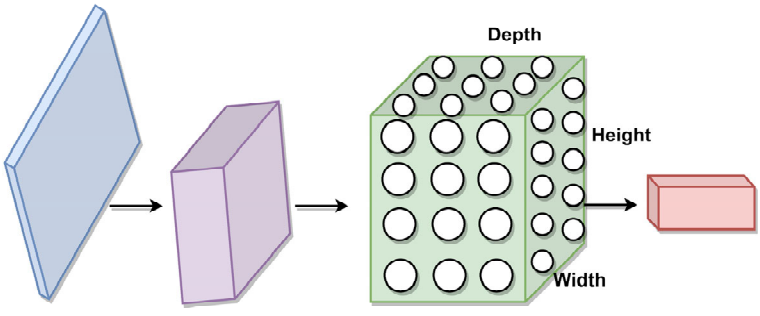
3.2 Why not BNN?

As stated above a BNN takes single vectors as inputs so a 2D image cannot fit in such a model. Moreover, a coloured image is not a 2D entity, it is 3D, since we have a third dimension of basic colours red, green, blue (RGB) which make coloured images so real. In general, a Canadian Institute for Advanced Research (CIFAR-10) image is used in CNN. A CIFAR-10 image has a dimension of 32 × 32 × 3 (i.e., 32 pixels each side and three shades). Also, if we consider making a 3D vector into a single vector, the size of the vector for a CIFAR image will be 32 × 32 × 3 = 3,072 parts along with a number of parameters for reconverting the vector back to a 3D matrix. For an image of size 256 × 256 × 3, the number climbs up to 196,608 for the first layer. This type of high numbers also gives rise to overfitting in a model which is discussed later. Hence, choosing a BNN is not a right choice for involving thousands of images.

3.3 3D volumes

A ConvNet model contains neurons in a three-dimensional fashion. The fact that a 3D image is an input forces the architecture to remain in its original form. The three dimensions of the ConvNet are named: width, height, depth (here, depth does not signify the full depth of the model). One layer has one three-dimensional filter (kernel) which slides on the whole dimension of the image and dot products to give a 3D volume. As the NN goes deeper from one layer to another the depth of the volume increases and the height and the width both decreases. Hence, the volume is a differentiated conclusion from the image after processing with the filters. The perception of this process is presented in Figure 2. In Figure 2, the model follows from left to right. It can be observed that the depth of volume is increasing as the model goes deeper and the height and width decrease in parallel.

Figure 2 The three-dimensional arrangement of CNN model (see online version for colours)



3.4 ConvNets layers

The ConvNet layers are basically divided into three parts, the input layer, hidden layer and the output layer. The hidden layer does all the work, and the work of input and output layers can easily be concluded. The input layer takes the input data (in form of image) and the output layer gives out the result. The hidden layer which is the most important for researchers and classified as the Conv layer, Conv stands for convolution. It contains the activation function of developer's choice [here, a rectified linear unit (ReLU) is used], the pooling layer, the FC layer.

A simple ConvNet model for CIFAR-10 images can have the following arrangement [*INPUT -> CONV -> ReLU -> POOL -> FC*]. The short description about each of them is as follows:

- *INPUT layer* contains pixel values in a matrix form for the image dimensions (for a CIFAR-10 image [32 × 32 × 3]).
- *Convolution layer* does the dot product between the receptive field of the input image and weights in the form of filters (kernels). The process is done by a sliding filter which slides all over the image to perform dot product with the filters. As the number of filters increases the depth of output volume increases.

- *ReLU* stands for rectified linear unit, ReLU is an activation function for each element in the volume. The function can be written as max(0, x), i.e., it does not pass negative values.
- *POOL layer* performs a process called pooling, which is decreasing the size of the volume, basically it decreases the height and the width. The function used is called MAX pooling which will be explained later.
- *FC layer*, as the name suggests is completely connected to all the values in the previous volume and helps to come to a desired conclusion. For example, for a CIFAR-10 sample, the output will be classified in ten classes (as CIFAR-10 dataset contain different images of ten classes such as car, ship, etc.) and hence the output might be percentage scores of all the classes.

So, from the details, it can be concluded that ReLU and pool layer are predefined functions. The main layers we had to work on are Conv layer and FC layer. It can be seen that the Conv layer contains a filter or kernel which transforms the input to required output and a FC layer is the final layer of the model and should produce a binary result for declaring an image forged or not. Hence, a FC layer can be a normal neuron and with proper activation function to declare the output.

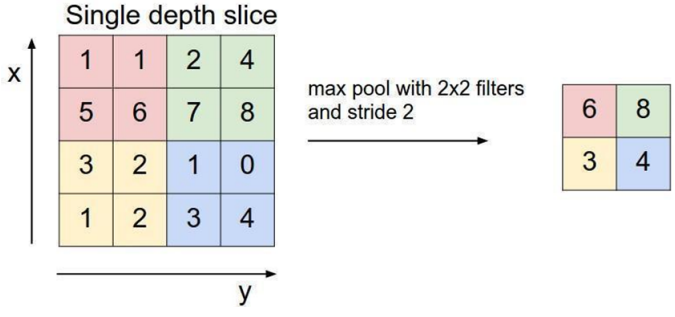
3.5 Important findings sofar

- A ConvNet model contains different layers to transform the input volume into final classes probabilities.
- Hidden layer contains different types of layers such as Conv layer, ReLU, pool layer and FC layer.
- Conv layer, ReLU, pool layer takes a 3D input volume as input and gives a 3D output volume.
- ReLU and the pool layer do not contain any parameters (filter/kernel).
- Hyperparameters, on which the learning rate depends, are not there in a ReLU layer.

3.6 Pooling layer

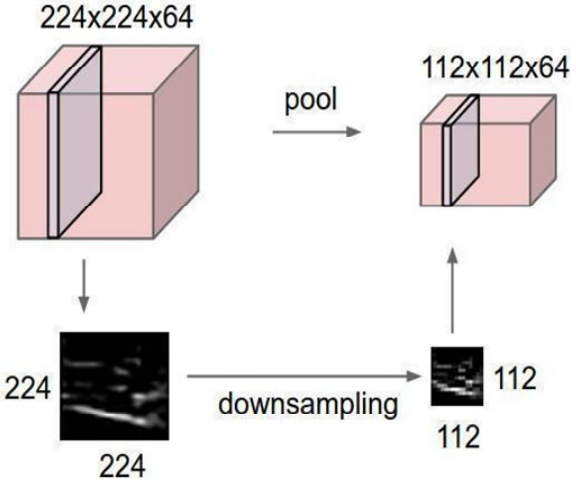
Pooling layer is occasionally embedded in the middle of progressive Conv layers during ConvNet design. It is used to dynamically decrease the spatial size of the portrayal to reduce the measure of boundaries and calculation in the system. This layer works freely on each solidity to resize the information spatially by employing the MAX activity. The poling layer having filter size 2 × 2 is a widely used structure with the stride of 2. The process of MAX pooling with filter size 2 × 2 and stride of two is represented in Figure 3. As per the MAX activity, MAX pooling takes the maximum number from the 2 × 2 filter region. From Figure 3, it can be observed that MAX pooling contributes to 75% reduction in slice's size. The pooling activity in case of 3D volume is represented in Figure 4. From Figure 4, it is worth to notice that the profundity measurements after pooling activity remain unchanged (i.e., only height and width decreases and depth remains unaltered).

Figure 3 An illustration of MAX pooling activity with filter size 2 × 2 and a stride of 2 (see online version for colours)



Source: Adopted from CS231n Convolutional Neural Networks for Visual Recognition

Figure 4 An illustration of MAX pooling activity for 3D volume (see online version for colours)



Source: Adopted from CS231n Convolutional Neural Networks for Visual Recognition

The general process for pooling layer is described below:

- Initialisation: width = W, height = H and depth = D.
- Receive a volume of size $W1 \times H1 \times D1$.
- Call two hyperparameters:
  1 spatial degree F
  2 stride S.
- Regenerate a volume of size $W2 \times H2 \times D2$ where the respective values are as follows:
  1 $W2 = (W1 - F) / S + 1$

2 $H2 = (H1 - F) / S + 1$

3 $D2 = D1$.

- Introduces zero boundaries since it processes a fixed capacity of the information.
- For pooling layers, rarely to cushion the info utilising zero-cushioning.

There are two types of maximum pooling, a pooling with $S = 3$, $F = 2$ also termed as covering pooling and pooling with $F = 2$, $S = 2$. Pooling with larger open fields is excessively devastating.

3.7 Completely associated (FC) layer

As found in the normal NN, in FC layers neurons have full associations with all initiations in the past layer. Their initiations can subsequently be registered with a lattice increase which is followed by an inclination counterbalance.

Why use padding: As discussed earlier, the advantage of keeping the spatial sizes fixed after CONV improves execution to a great extent. If there is a chance that the CONV layers will not protect the data sources to zero and only perform significant turns, the size of the volumes would scarcely decrease after each CONV. At this point, the data at the edges would be 'washed away' too quickly.

3.8 Computational considerations

The considerable bottleneck to know about while developing ConvNet models is the memory bottleneck. Even after enormous advancement in the GPUs, there is still a constraint of 3/4/6 GB memory. The latest GPU has about 16 GB of memory which is very costly to effort. There are three significant wellsprings of memory to monitor:

- From the halfway volume measures: These are the obscene actuations at each layer of the ConvNet, and their slopes (of equivalent size). Normally, most of the depictions are on the prior layers of a ConvNet. These are kept around because they are required for backpropagation. However, a smart usage that runs a ConvNet just at test time could on a basic level minimise this by an enormous sum, by just putting away the present presentations at any layer and disposing of the past actuations on layers underneath.
- From the boundary estimates: These are the numbers that hold the system boundaries. Their inclinations during backpropagation, and ordinarily likewise a stage reserve if the enhancement is utilising force. In this way, the memory to store the boundary vector alone should normally be duplicated by a factor of in any event 3 or somewhere in the vicinity.
- Every convent execution needs to keep up various memories, for example, the picture information, clusters, their increased variants, and so on.

When you have an unpleasant assessment of the complete number of qualities (for enactments, slopes and misc), the number must be changed over to estimate in GB. Take the quantity of qualities, duplicate by 4 to get the crude number of bytes (since each coasting point is 4 bytes, or perhaps by 8 for twofold accuracy), and afterward partition

by 1,024 on numerous occasions to get the measure of memory in KB, MB, lastly GB. On the off possibility that your system does not fit, a typical heuristic to 'make it fit' is to diminish the cluster size, since a large portion of the memory is generally devoured by the actuations.

4 Error level analysis

ELA awards recognising districts inside an image that are at different weight levels. With JPEG pictures, the entire picture should be at commonly a comparative level. In case a zone of the image is at a generally one of a kind blunder level, by then it likely shows a propeller change. ELA highlights the contrasts in the JPEG pressure rate. Areas with uniform concealing like a solid blue sky or a white divider will presumably have a lower ELA result (darker concealing) than high-separate edges. There are following things to search for:

- Edges: Analogous edges should have equivalent brilliance in the ELA result. Each highly distinguished edge must appear to be like one another, and all low-differentiate edges should appear to be comparative. With a unique photograph, low-differentiate edges ought to be nearly as brilliant as high-differentiate edges.
- Textures: Similar surfaces ought to have comparable shading under ELA. Zones with increasingly surface detail, for example, a nearby b-ball, will probably have a better ELA ensue than a smooth out surface.
- Surfaces: Irrespective of the genuine shade of the surface, every level surface ought to have about a similar shading under ELA.

Check out the image and recognise the diverse high-differentiate edges, low-differentiate edges, surfaces and surfaces. Contrast those zones and the ELA results. On the off chance that there are noteworthy contrasts, at that point, it distinguishes dubious regions that might have been carefully modified. Resaving a JPEG image leaves high-frequencies and scores less contrasts between high-differentiate edges and surfaces. A poor-quality JPEG image will appear as dull. Scaling an image littler can support high-differentiate edges, making them more brilliant under ELA. Correspondingly, sparing a JPEG with an Adobe item will consequently polish high-differentiate edges and surfaces, causing them to show up a lot more splendid than low-surface sides.

4.1 Lossy and lossless

There are a wide range of picture document groups. A few arrangements are lossy, while others are lossless.

- *Lossless:* Lossless record designs hold careful pixel shading data. On the off chance that you load an image, spare it, and burden it once more, every pixel will have precisely the same value. Indeed, even a change between two lossless configurations will hold precisely the same shading values. For instance, portable network graphics (PNG) and bitmap (BMP) are two diverse lossless configurations. On the off chance that you convert an image from a BMP to a PNG, it will hold precisely the same pixel values, even though the document position changed.

- *Lossy:* A lossy record design does not ensure that the shades will remain the equivalent. With JPEG, sparing requires indicating a quality level. The quality level alters the pressure sum (lower quality makes littler documents), however it packs by expelling some shading data. With JPEG, sparing an image makes the shades change a bit. The resaved record may outwardly look equivalent to the source image; however, the specific pixel esteems to be contrast.

With lossy picture arrangements, the first run through a picture is spared and causes a lot of shading misfortune. Be that as it may, stacking the image and afterward encoding it again in the equivalent lossy arrangement will bring about less extra shading debasement. The ELA results feature the territories in the picture that are generally inclined to shading degradation during a resave. Alters commonly stand apart like a district with a higher corruption potential contrasted with the remainder of the picture.

4.2 ELA of lossy images

JPEG pictures utilise a lossy pressure framework. Every re-encoding (resave) of the picture adds greater quality misfortune to the picture. In particular, the JPEG calculation works on an 8 × 8 pixel framework. Each 8 × 8 square stays packed. If picture is totally unmodified, at that point every 8 × 8 square ought to have comparative blunder possibilities. If the picture is unmodified and resaved, at that point each square ought to debase at roughly a similar rate. ELA spares the picture at a predefined JPEG quality level. This resave presents a known measure of blunder over the whole picture. The resaved picture is then looked at against the first picture.

If a picture is changed, at that point each 8 × 8 square that was moved by the alteration ought to be at a higher blunder prospective than the remainder of the picture. Altered territories will show up with a greater possible mistake level. At the point when a document is changed over from a lossy record organisation to a lossless configuration, resave curios is held. This licenses ELA to feature changes made to a JPEG picture that was changed over to PNG. The all lossy document designs are not good. For instance, the lossy WebP, HEIC, and HD Photo (JPEG XR) positions utilise distinctive pressure calculations than JPEG. Regardless of whether a JPEG picture has been over and again spared, it might at present outcome in first time-spared curios, if it is changed over to these other document groups. This happens on the grounds that the WebP, HEIC, and HD Photo ancient rarities are applied.

4.3 ELA for lossless images

Lossless document positions do not change the hues in an image. At the point, as a lossy image is changed into a lossless arrangement, the entirety of the lossy artefacts is held. This grant recognising explicit sorts of adjustments, for example:

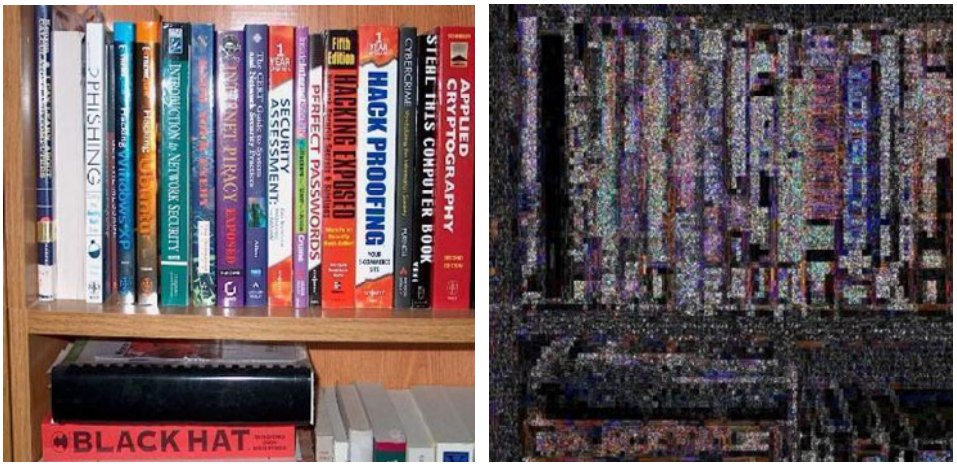
- File group transformations: Switching from JPEG into PNG will hold the past JPEG antiquities. Since a local PNG ought not include JPEG ancient rarities, the change is noticeable.
- Native lossless: If an image has never experienced a JPEG encoding, it will never have JPEG antiquities. ELA must report a steady picture quality, and no 8 × 8 or

16 × 16 matrix-based obstructing, since it speaks to the regions that will change during the first JPEG encoding.

4.4 Assessing ELA

Along with ELA, each matrix that is not upgraded for the quality level will demonstrate lattice squares which have been changed during a resave. For instance, advanced cameras do not enhance pictures for predetermined camera quality level (high, medium, low and so forth.). Figure 5 shows a JPEG image [Figure 5(a)] and its ELA [Figure 5(b)]. Let us consider this image as original image, from Figure 5, it can be observed that solid colour having minimum ELA (darker) and the mixed colours having maximum ELA (whiter). Unique pictures from computerised cameras must get a high level of progress during any resave (high ELA esteems). Each ensuing resave will bring down the blunder level potential, yielding a darker ELA result. This process has been depicted in Figure 6. It can be observed that after a resave the ELA values have been reduced. If this image is further resaved the ELA will become darker. With enough resaves, the network square will in the end arrive at its base blunder level, where it will not change any longer.

Figure 5 (a) Original JPEG image (b) ELA of the original JPEG image (see online version for colours)

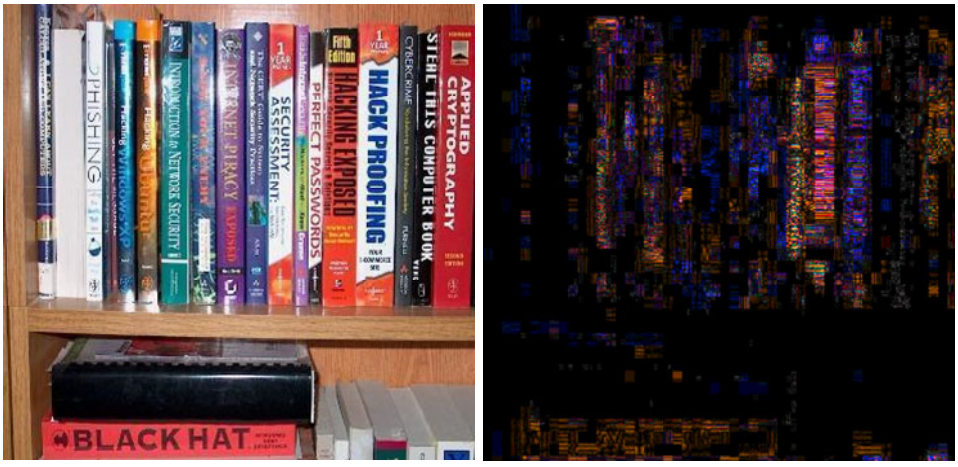


(a) (b)

Source: Adopted from Fotoforensics (http://fotoforensics.com/tutorial-ela.php)

Recognise that high recurrence territories such as edges and objects as a rule have elevated ELA esteems than the rest of the image. For instance, the content on the books stands apart on the grounds that the light/dull differentiation makes a high recurrence edge. The edited image with ELA is represented in Figure 7, the editing in the original image is clearly visible in ELA. The adjusted toy and some books can be figured out easily due to maximum ELA. It can be concluded that there are contrast edges and edges and surfaces with surfaces. If all surfaces except for one have comparable ELA esteems, at that point the anomaly ought to be suspect.

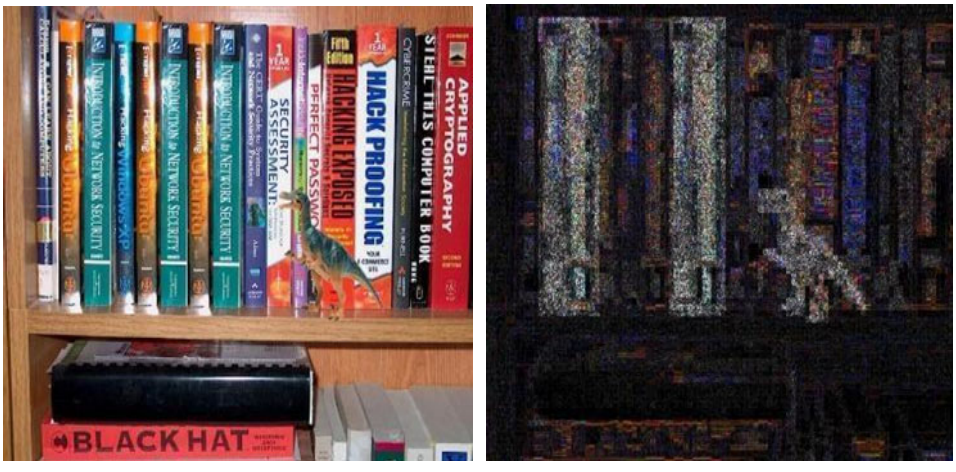
Figure 6 (a) Once resaved image of the original JPEG image (b) ELA of the resaved image (see online version for colours)



(a) (b)

Source: Adopted from Fotoforensics (http://fotoforensics.com/tutorial-ela.php)

Figure 7 (a) Image adjusted with a toy and some books (b) The ELA of the adjusted image (see online version for colours)



(a) (b)

Source: Adopted from Fotoforensics (http://fotoforensics.com/tutorial-ela.php)

4.5 Advanced uses

With preparing and practice, ELA clients can likewise figure out how to recognise picture scaling, quality, editing, and resave changes. For instance, if a non-JPEG picture comprises noticeable network boundaries (one-pixel wide in 8 × 8 squares), at that point,

it implies the image began as a JPEG and was changed over to the non-JPEG design. On the off chance that a few zones of the image need framework lines or the lattice lines move, at that point it means a join or attracted partition the non-JPEG picture. Another model, a PNG record is a lossless document design. If an image is a unique PNG, at that point ELA should create high qualities for edges and surfaces. Nonetheless, if ELA produces feeble outcomes (dim or dark shading) along edges and surfaces, at that point the PNG is expected to be made from a JPEG. This is on the grounds that the change procedure from JPEG to PNG is lossless and will hold JPEG curios. At the point when joined with different calculations, ELA turns into an exceptionally incredible assessment instrument.

4.6 Provisions for ELA

While ELA is a superb device for identifying adjustments, there are various admonitions:

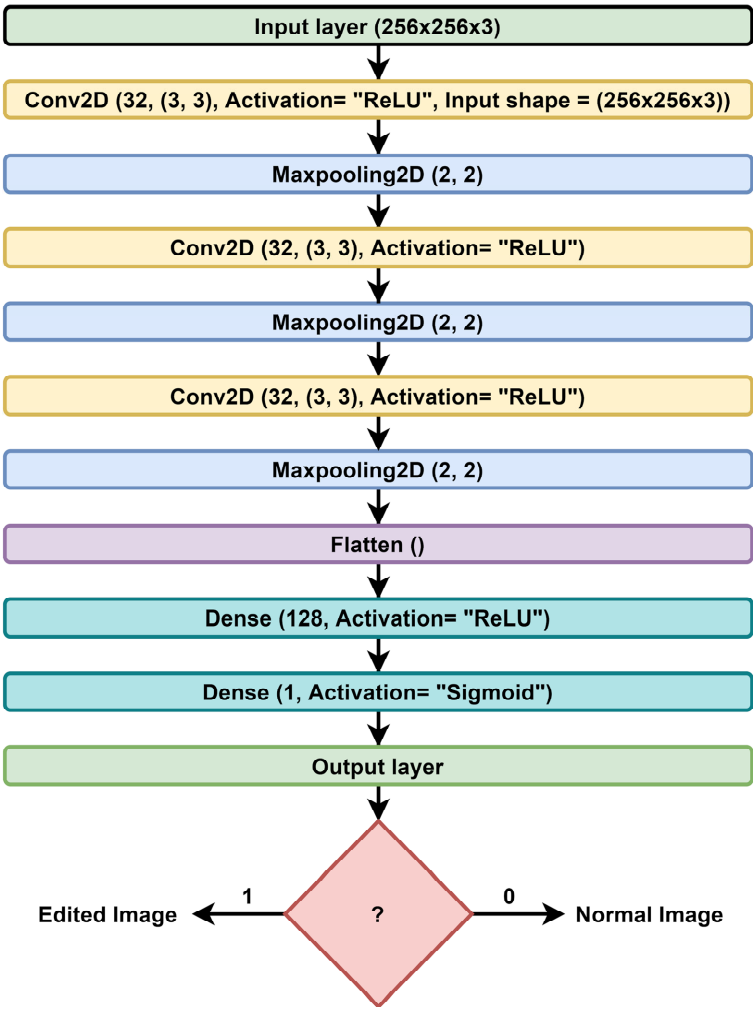
- A pixel change, or slight shading modification, may not produce a recognisable change in the ELA.
- JPEG works on a framework, a modification to every piece of the lattice will probably affect the whole matrix square. One will be unable to distinguish precisely which pixel in the network was changed.
- JPEG utilises the YUV shading space. High-level difference hues in a similar lattice, for example, highly contrasting, orange, and blue, or green and will as a rule create higher ELA esteems than comparative hues in a similar network.
- ELA just recognises what districts have distinctive pressure levels. It does not distinguish sources. On the off chance that a lower quality picture is joined into a more excellent picture, at that point the poorer quality picture might show up as a bleaker locale.
- Recolouring, scaling, or adding clamour to a picture will alter the whole picture, making a higher mistake level potential.
- If a picture is resaved on different occasions, at that point, it might be altogether at the very least blunder level. For this situation, the ELA will restore a dark picture and no adjustments can be distinguished utilising this calculation.
- With Photoshop, the basic demonstration of sparing the image can auto-hone surfaces and edges, making a higher mistake level potential. This antiquity does not distinguish slow change; it recognises that an Adobe item was utilised. Technically, ELA shows up as an alteration since Adobe consequently played out a change, however the adjustment was not deliberate by the client.

ELA is just a single calculation. The understanding of results might be uncertain. It is critical to approve discoveries with different investigation methods and calculations.

5 Results and discussion

The flowchart of the proposed CNN model for image forgery detection is shown in Figure 8. The model consists of the multilayer ConvNet with more than one convolution, ReLU and pooling layers.

Figure 8 Flowchart of proposed CNN model (see online version for colours)



The proposed model is trained with a Kaggle dataset named CASIA1. The CASIA1 is further divided into two parts: dataset 1 and dataset 2. During the testing phase, the input layer has been fed with image with the size of 256 × 256 × 3. The image goes under different layers of multilayer ConvNet. This model uses 'ReLU' as an activation function and MAX pooling as pooling activity. In the output layer to obtain the binary decision about the image is forged or not the simple activation function 'sigmoid' is applied. After

obtaining the binary output from a binary decision has been applied. If the output is '1', the image is forged, otherwise a normal image.

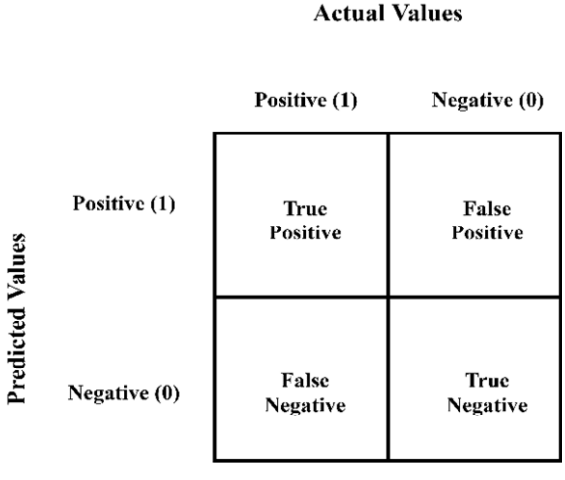
5.1 Performance measures

5.1.1 Confusion matrix

In the vast area of AI, there is an issue of accurate understanding. The accuracy of the system should be up to the mark otherwise it will be called as a blunder framework. For this analysis, a table is designed that permits representation of the performance of a calculation termed as confusion matrix. The general form of the confusion matrix is presented in Figure 9, it has following parts:

- *True positives (TP):* These are the effectively anticipated positive qualities which imply that the estimation of real class is yes, and the estimation of anticipated class is additionally yes.
- *True negatives (TN):* These are the accurately anticipated negative qualities which imply that the estimation of real class is no, and estimation of anticipated class is additionally no.

Figure 9 A confusion matrix

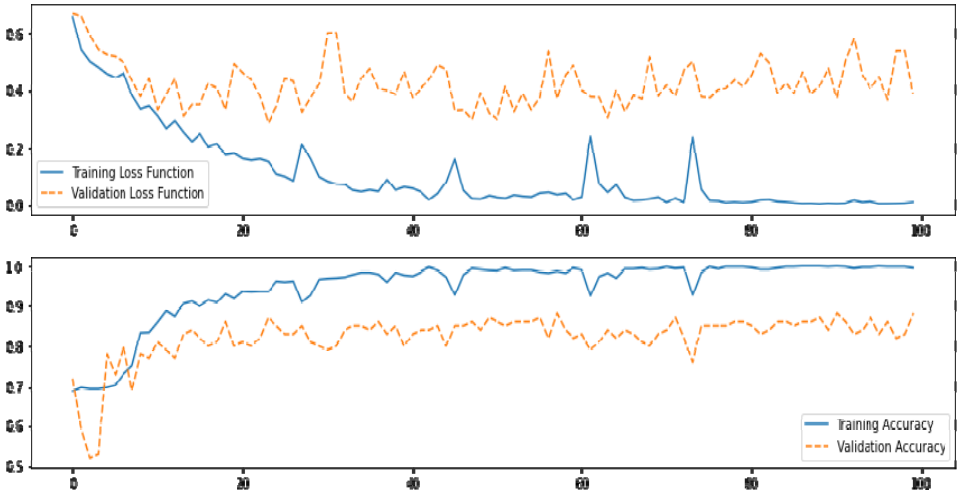


Bogus positives and bogus negatives, these qualities happen when your genuine class repudiates with the anticipated class.

- *False positives (FP):* When genuine class is no, and anticipated class is yes.
- *False negatives (FN):* When genuine class is yes, yet anticipated class in no.

On training the network with two datasets obtained from Kaggle website (CASIA1), the graphs of loss functions (LFs) and accuracy are described below for both training and validation. The network training and validation has been performed with Google Colab notebook. Figure 10 shows training loss and accuracy for dataset 1. In Figure 10, the *Y*-axis represents the normalised accuracy and *X*-axis defines number of epochs.

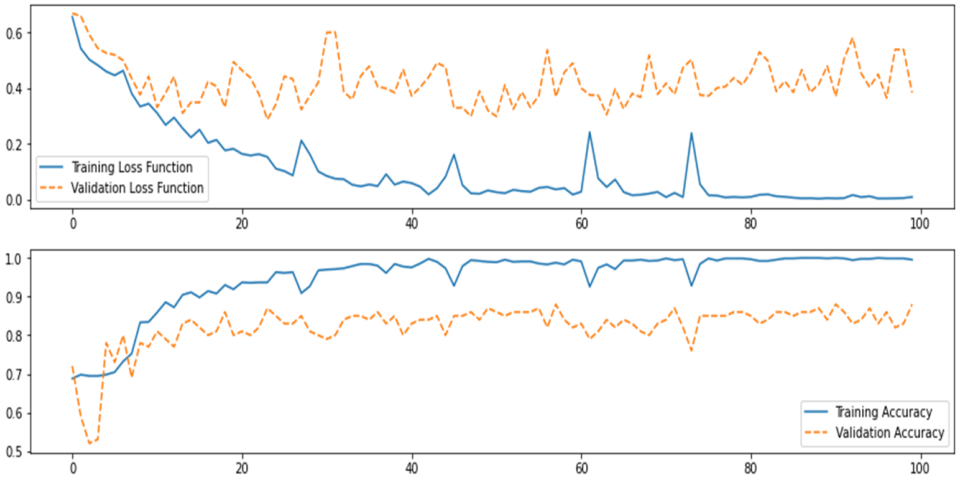
Figure 10 Training losses and accuracy for dataset 1 (see online version for colours)



From Figure 10, it can be deduced that the proposed model has achieved around 99% accuracy during the tanning process and around 91% accuracy during validation process. The complete simulation models have been tested for 100 epochs.

The outcomes of the dataset 2 with respect to LF and accuracy have been presented in Figure 11. The simulation setup for dataset 2 is the same as discussed in earlier case. With dataset 2 also, the proposed model shows the stable performance with the image forgery detection accuracy of around 87%. Thus, the simulation results validate the proposed model for image forgery detection has an accuracy of around 90%.

Figure 11 Training losses and accuracy for dataset 2 (see online version for colours)



For the proposed ConvNet model, the confusion matrix of test dataset 1 and dataset 2 is represented by Figure 12 and Figure 13, respectively.

- *Accuracy of the proposed model:* Accuracy is the most instinctive exhibition measure in DL. It is essentially a proportion of effectively anticipated perception of the model. As per the confusion matrix, the normalised accuracy of ConvNet model is represented as follows:

  $$Accuracy = (TP + TN) / (TP + TN + FP + FN)$$

  The proposed model achieves 0.908 for dataset 1 which implies the model is approx. 91% accurate. For dataset 2, it is around 0.872 which implies the model is approximately 87% accurate.

- *Precision:* Precision is the proportion of effectively anticipated positive perceptions to the complete anticipated positive recognition. The precision of a system is given as follows:

  $$Precision = TP / (TP + FP)$$

  The model presented in this work shows precision of 0.985 for dataset 1, which is truly acceptable and 0.895 for dataset 2, which is acceptable.

- *Recall (sensitivity):* Recall is the proportion of accurately anticipated positive perceptions to all perceptions in genuine class – yes. The recall of the system can be given by:

  $$Recall = TP / (TP + FN)$$

  The proposed model has a recall of 0.82 for dataset1 and 0.80 for dataset 2 which is useful for this model.

- *F1 score:* F1 score is the weighted normal of precision and recall and given by:

  $$F1\ score = 2 \times (recall \times precision) / (recall \times precision)$$

  Consequently, this score considers both bogus positives and bogus negatives. Instinctively, it is not as straightforward as precision, yet F1 is normally more valuable than exactness, particularly if you have a lopsided class circulation. Exactness works best if bogus positives and bogus negatives have comparable expense. If the expense of bogus positives and bogus negatives are altogether different, it is smarter to take a gander at both precision and recall. For the model presented here, the F1 score is 0.89 for dataset 1 and 0.84 for dataset 2.

Figure 12 Confusion matrix for dataset 1 (see online version for colours)

197	3
43	257

Figure 13 Confusion matrix for dataset 2 (see online version for colours)

179	21
43	257

6 Conclusions

In this paper, a CNN model along with ELA has been presented for image forgery detection. ELA for image forgery detection has been discussed and demonstrated with suitable examples. The proposed model has been validated with Kaggle's CASIA1 dataset. It has been found that the model performs with 91% accuracy for dataset 1 and around 87% accuracy for dataset 2. The detailed efficacy of the system is measured based on the confusion matrix for both the datasets. It is concluded that the system has a normalised precision of 0.98 and 0.89 for dataset 1 and dataset 2, respectively. Further, the proposed model is tested for recall and F1 score. For all the performance measure criteria's, performance of the model was found satisfactory. Based on the propped model an android application can be developed in a future. The application will determine whether the image is digitally altered/edited or not.

Acknowledgements

Authors would like to thank anonymous reviewers for providing the constructive comments on this work and help us to improve the quality of this article.

References

Ansari, M.D., Ghrera, S.P. and Tyagi, V. (2014) 'Pixel-based image forgery detection: a review', *IETE Journal of Education*, Vol. 55, No. 1, pp.40–46.

Bayar, B. and Stamm, M.C. (2016) 'A deep learning approach to universal image manipulation detection using a new convolutional layer', in *Proceedings of the 4th ACM Workshop on Information Hiding and Multimedia Security*.

Bharti, C.N. and Tandel, P. (2016) 'A survey of image forgery detection techniques', in *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*.

Birajdar, G.K. and Mankar, V.H. (2013) 'Digital image forgery detection using passive techniques: a survey', *Digital Investigation*, Vol. 10, No. 3, pp.226–245.

Chen, J., Kang, X., Liu, Y. and Wang, Z.J. (2015) 'Median filtering forensics based on convolutional neural networks', *IEEE Signal Processing Letters*, Vol. 22, No. 11, pp.1849–1853.

CS231n Convolutional Neural Networks for Visual Recognition, Stanford University [online] https://cs231n.github.io/convolutional-networks/.

Das, D., Nayak, D.R., Dash, R., Majhi, B. and Zhang, Y-D. (2020) 'H-WordNet: a holistic convolutional neural network approach for handwritten word recognition', *IET Image Processing*, Vol. 14, No. 9, pp.1794–1805.

Doegar, A., Dutta, M. and Gaurav, K. (2019) 'CNN based image forgery detection using pre-trained AlexNet model', *International Journal of Computational Intelligence & IoT*, Vol. 2, No. 1, pp.1–6.

Dureja, A. and Pahwa, P. (2018) 'Image retrieval techniques: a survey', *International Journal of Engineering & Technology*, Vol. 7, Nos. 1–2, pp.215–219.

Farid, H. (2009) 'Image forgery detection', *IEEE Signal Processing Magazine*, Vol. 26, No. 2, pp.16–25.

Fineman, M. (2012) *Faking It: Manipulated Photography Before Photoshop*, Metropolitan Museum of Art.

Fotoforensics [online] http://fotoforensics.com/tutorial-ela.php (accessed 05 October 2021).

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J. et al. (2018) 'Recent advances in convolutional neural networks', *Pattern Recognition*, Vol. 77, pp.354–377 [online] https://www.sciencedirect.com/journal/pattern-recognition /issues.

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S. and Lew, M.S. (2016) 'Deep learning for visual understanding: a review', *Neurocomputing*, Vol. 187, pp.27–48 [online] https://www.sciencedirect.com/journal/neurocomputing/issues.

Hacker Factor [online] http://www.hackerfactor.com/ (accessed 05 October 2021).

Han, H., Li, Y. and Zhu, X. (2019) 'Convolutional neural network learning for generic data classification', *Information Sciences*, Vol. 477, pp.448–465 [online] https://www.sciencedirect.com/journal/information-sciences/issues.

Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) 'Imagenet classification with deep convolutional neural networks', in *Advances in Neural Information Processing Systems*.

Liu, S. and Deng, W. (2015) 'Very deep convolutional neural network based image classification using small training sample size', in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*.

Lopes, P.A., Yadav, S.S., Ilic, A. and Patra, S.K. (2019) 'Fast block distributed CUDA implementation of the Hungarian algorithm', *Journal of Parallel and Distributed Computing*, Vol. 130, pp.50–62 [online] https://www.sciencedirect.com/journal/journal-of-parallel-and-distributed-computing/issues.

Mishra, P., Mishra, N., Sharma, S. and Patel, R. (2013) 'Region duplication forgery detection technique based on SURF and HAC', *The Scientific World Journal*, Vol. 2013, Article ID 267691, 8pp., https://doi.org/10.1155/2013/267691.

Mushtaq, S. and Mir, A.H. (2014) 'Forgery detection using statistical features', in *2014 Innovative Applications of Computational Intelligence on Power, Energy and Controls with Their Impact on Humanity (CIPECH)*.

Nayak, D.R., Dash, R. and Majhi, B. (2020) 'Automated diagnosis of multi-class brain abnormalities using MRI images: a deep convolutional neural network based method', *Pattern Recognition Letters*, Vol. 138, pp.385–391 [online] https://www.sciencedirect.com/journal /pattern-recognition/issues.

Popescu, A.C. and Farid, H. (2005) 'Exposing digital forgeries in color filter array interpolated images', *IEEE Transactions on Signal Processing*, Vol. 53, No. 10, pp.3948–3959.

Selamat, S.R., Yusof, R. and Sahib, S. (2008) 'Mapping process of digital forensic investigation framework', *International Journal of Computer Science and Network Security*, Vol. 8, No. 10, pp.163–169.

Shwetha, B. and Sathyanarayana, S. (2016) 'Digital image forgery detection techniques: a survey', *ACCENTS Transactions on Information Security*, Vol. 2, pp.22–31.

Uliyan, D.M., Jalab, H.A., Wahab, A.W.A., Shivakumara, P. and Sadeghi, S. (2016) 'A novel forged blurred region detection system for image forensic applications', *Expert Systems with Applications*, Vol. 64, pp.1–10 [online] https://www.sciencedirect.com/journal/expert-systems -with-applications/issues.

Walia, S. and Kumar, K. (2019) 'Digital image forgery detection: a systematic scrutiny', *Australian Journal of Forensic Sciences*, Vol. 51, No. 5, pp.488–526.

Yadav, S.S., Lopes, P.A.C., Ilic, A. and Patra, S.K. (2019) 'Hungarian algorithm for subcarrier assignment problem using GPU and CUDA', *International Journal of Communication Systems*, Vol. 32, No. 4, p.e3884.

Zhang, J., Zhu, W., Li, B., Hu, W. and Yang, J. (2016a) 'Image copy detection based on convolutional neural networks', in *Chinese Conference on Pattern Recognition*.

Zhang, Y., Goh, J., Win, L.L. and Thing, V.L. (2016b) 'Image region forgery detection: a deep learning approach', *SG-CRC*, Vol. 2016, pp.1–11.

Zhou, J., Ni, J. and Rao, Y. (2017) 'Block-based convolutional neural network for image forgery detection', in *International Workshop on Digital Watermarking, Lecture Notes in Computer Science*, Vol. 10431, Springer, Cham., https://doi.org/10.1007/978-3-319-64185-0_6.