Image Manipulation Detection-A Review

[1]Sanket Sahay, [2]Kunal Poladia, [3]Udita Talsaniya

[1]Student of B.E.(EXTC), [2] Student of B.E.(EXTC), [3] Student of B.E.(EXTC)
[1]Thakur College of Engineering and Technology,
[11]Thakur College of Engineering and Technology, Mumbai, India

Abstract : This paper is a review of Image Manipulation Detection technology. It aims to study the previous work in image manipulation detection, what algorithms and what algorithms currently in use for other purposes can be used for image manipulation detection. The papers reviewed here studied JPEG image compression artifacts, noise profiles, and different convolutional neural networks and these were used to detect if the given image was manipulated and region of manipulation.

IndexTerms **- Component,formatting,style,styling,insert.**

I. INTRODUCTION

In the age of the internet, where information spreads rapidly, forgery is a huge concern for safety of property and life. The increase in use of multimedia for information consumption over the ages makes it a vulnerable and obvious target.

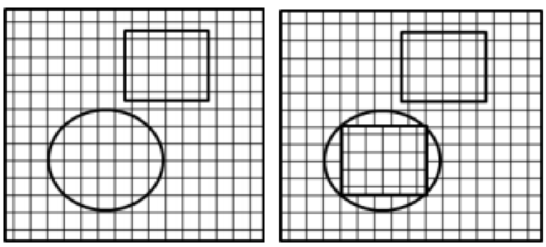
Here we study a total of six papers that tackle this issue in different ways. The first two papers use artefacts introduced in the image due to manipulation in terms of changes in the blocking artefact grids introduced by JPEG and noise inconsistencies respectively. The next two papers use Convolutional Neural Networks to extract manipulation information from the input images that point to manipulation. These networks can be trained to detect these anomalies in the manipulated images. Further, we study two computer vision algorithms of object detection, namely Fast R-CNN and Faster R-CNN. These are very successful object detection algorithms and the hope is to study if its possible to tweak these algorithms to look for manipulation anomalies in an image instead of objects.

II. REVIEWED PAPERS

2.1 JPEG Image Tampering Detection Using Blocking Artefacts

Dijana Tralic et al [1], this project focuses on Image Manipulation Detection in compressed image formats such as JPEG.

This paper proposes a method to detect image manipulation by detecting misalignments in the Blocking artefacts introduced in the image by JPEG compression. The JPEG image compression is a lossy compression technique. It separates the image into 8 by 8 pixel blocks. DCT is performed on each block individually and then multiplied by a predefined quantization table. This image (after applying IDCT to the 8 by 8 pixel blocks), due to losses, will have blocky artefacts, especially in highly textured areas. This forms a grid of anomalies that is called blocking artefact grid (BAG). When a JPEG compressed image is copy-pasted over another JPEG image, it is highly likely that the BAGs of the two images will not align. This paper proposes a method to detect this mismatching of BAGs to detect the manipulated region of the image.



(a) Original image (b) Doctored image

The above image shows the original image BAG and the mismatched BAGs in the image after copy-paste.

This mismatching of the BAGs is done by analysing the high frequency AC coefficients of the image. When applying DCT to an 8 by 8-pixel block, the low frequency energy is shoved into the top left DC coefficient and the resulting AC coefficients therefore contain lower values. The AC coefficients after quantization are all zero or very close to zero. If AC coefficients are greater than zero, this signifies BAG mismatch. A local noise effect of an 8 by 8 block is calculated for locating blocking artefacts. Local effect is given by the formula:

$$LE = \sqrt{\left(\frac{\sum_{i=8||j=8} S_{ij}^2}{S_{11}^2}\right)}$$

where, S_{ij} represents the AC coefficients in the 8 by 8 block. AC coefficients are all the values in the rightmost columns and lowermost rows.

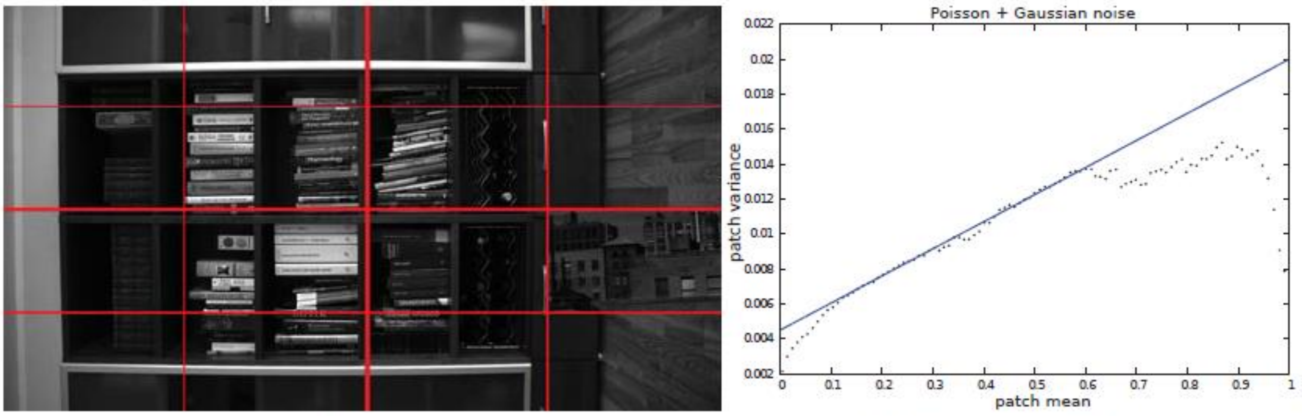
An 8 by 8 window is slid throughout the image and the local effect of every window is calculated. These local are analysed to find out mismatched BAGs.

2.2 Automated Image Splicing Detection from Noise Estimation in Raw Images

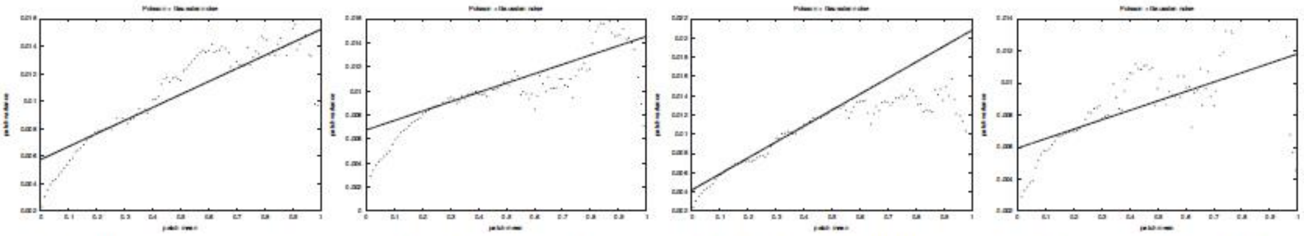
Thibaut Julliand el al [2], this project focuses on image manipulation detection on RAW image format files.

This paper proposes a method to detect manipulation of an image done by splicing by detecting local noise inconsistencies. This paper focuses on Gaussian noise and Poisson noise. The Gaussian and Poisson noise of an image depends on the lighting of the scene captured in the image and the internal electronic noises of the sensor. This results in different images having very unique noise profiles. When a portion from another image is spliced into a given image, the noise profile is generally different. It is very difficult to mask this difference in local noise inconsistencies.

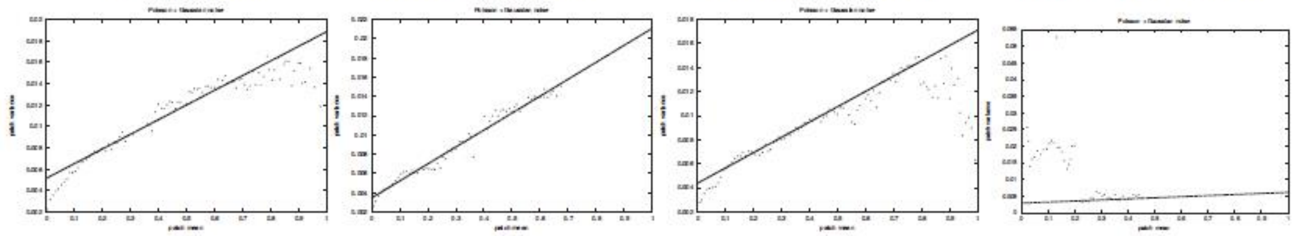
To find the local noise profile inconsistencies, the following method is used. The noise estimation is done by plotting the Poisson+Gaussian noise on a patch mean vs. patch variance plane. First the noise estimation for the whole image is plotted. Then the Image is separated into 4 equal regions. The noise estimation for these four images are plotted. Then The image is further divided into 16 equal regions. The noise estimations for all these regions are plotted.



(a) Input image (b) full image noise estimation



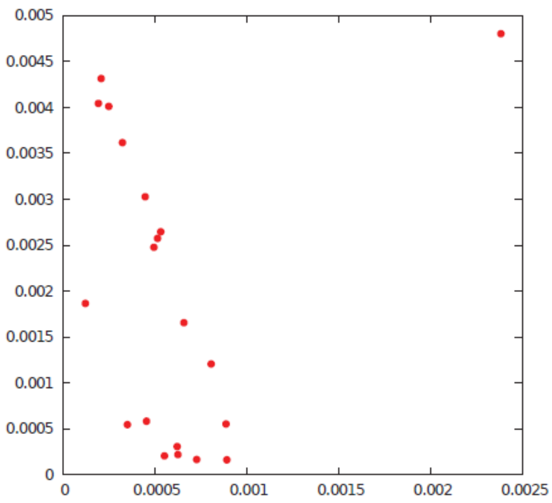
(c) Top-left quarter (d) Top-right quarter (e) Bottom-left quarter (f) Bottom-right quarter

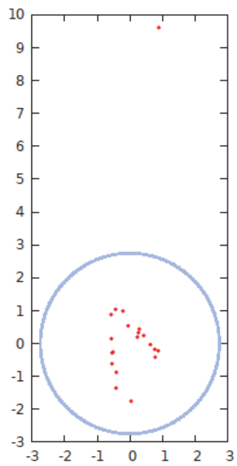


(g) Some 1/16 of the image (h) Some 1/16 of the image (i) Some 1/16 of the image (j) The spliced zone

Above image shows the image divided into quadtree and some of the noise estimations plotted.

The noise profile of each region is plotted on a Gaussian vs. Poisson plane. Here, each point represents a region of the image. After applying Robust Principle Component Analysis, clustering is performed. The point furthest to the other points represents the region with the most different noise profile and hence is deemed to be the edited region. The thresholding of clustering is applied as twice of the mean distance between all the points. Any point found to be further than this distance is considered to belong to the edited region.





The above Image shows Gaussian Vs, Poisson plane after robst Principle component analysis and also shows clustering.

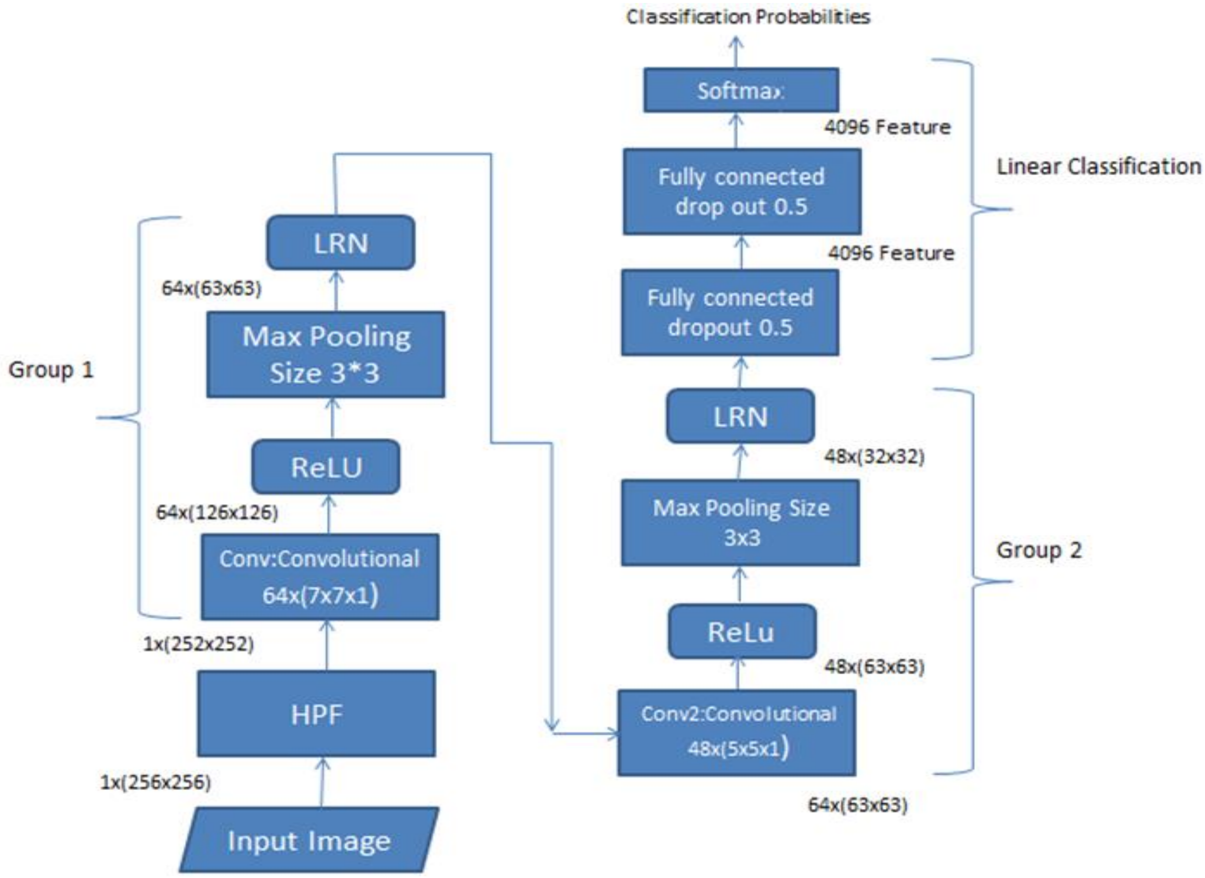
2.3 Image manipulation detection using convolution neural network

Dong-Hyun Kim 1 and Hae-Yeoun Lee, [3] Department of computer software Engineering, Kumoh National Institute of Technology proposed a CNN-based deep learning model for detecting image manipulation.

In this paper, convolution neural network is applied on the image. High pass filter is applied to extract hidden features. Manipulated image is generated by manipulation of original image. Manipulation is done by applying Gaussian filtering, median filtering, additive white Gaussian noise. The convolution layer comprises of maximum pooling, ReLU activation, local response normalization.

In this deep learning approach is applied for image manipulation detection. The process is divided in two parts learning and testing. For learning manipulated image is used. Manipulated image is generated by manipulation of original image. Manipulated image is classified into learning set and testing set. Then, learning set of manipulated image is fed to the designed CNN model. This CNN model consist of one high pass filter, 2 convolution layer, 2 fully-connected layer and one output layer.

Weights are updated using back propagation. After learning, test set is fed to the learned model and accuracy of proposed model is calculated.



2.4 A Deep Learning Approach to Universal Image Manipulation Detection using a New Convolution Layer

Belhassen Bayar and Matthew C. Stamm, [4] Drexel University, Department of ECE, USA in june2016 implemented CNN using the Caffe deep learning framework.

In this paper, Universal forensic approach is applied to detect manipulation of image. Specifically it uses new convolution layer. In this, manipulation of image is automatically done from training set. The new convolution layer is designed to suppress an image content and is capable of detecting multiple image manipulation without relying on pre-defined features of model.

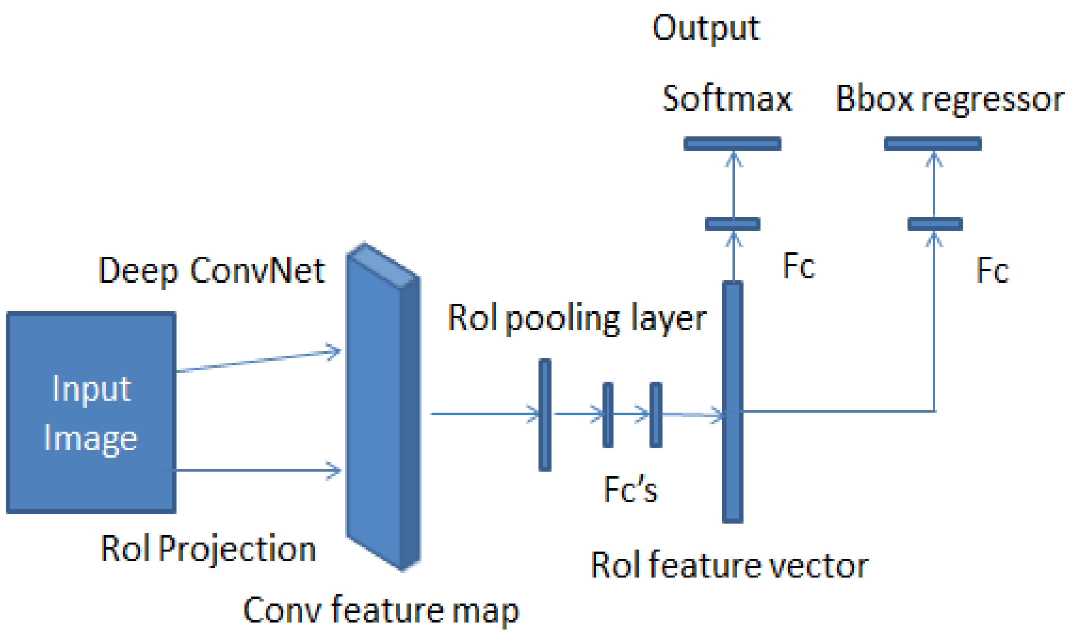
The convolution layer is applied to local regions with overlapping distance called strides. A convolution layer is followed by nonlinear mapping applied in an activation layer. To reduce the resolution of feature map max pooling is used. Then central value in neighbourhood is normalized by using Local response normalization.

They used Nvidia GeForce GTx 980 GPU with 4GB RAM. They converted datasets to lmdb format. It uses CNN model as a binary and multi-class classifier. Experimental datasets have been collected from 12 different camera models with no previous tampering or preprocessing. Median, Gaussian blurring, Additive White Gaussian Noise (AWGN) and Resampling operations are performed on unaltered images. This proposed model achieves an accuracy of more than 95% of detecting the different four type of forgery.

2.5 Fast R-CNN

In this paper Ross Girshick from Microsoft research have proposed a fast region based r-cnn network for object detection.

It employs several innovations in training and testing speed while also increasing the accuracy. This paper proposes single-train algorithm that propose the location of the image and classifies the image in spatial domain. The runtime of detection network process image is 0.3 sec with achieving accuracy of 66%. R-CNN is slow without sharing computation because it performs ConvNet for each object proposal. SPnet is use which 10 times faster than and more accurate.



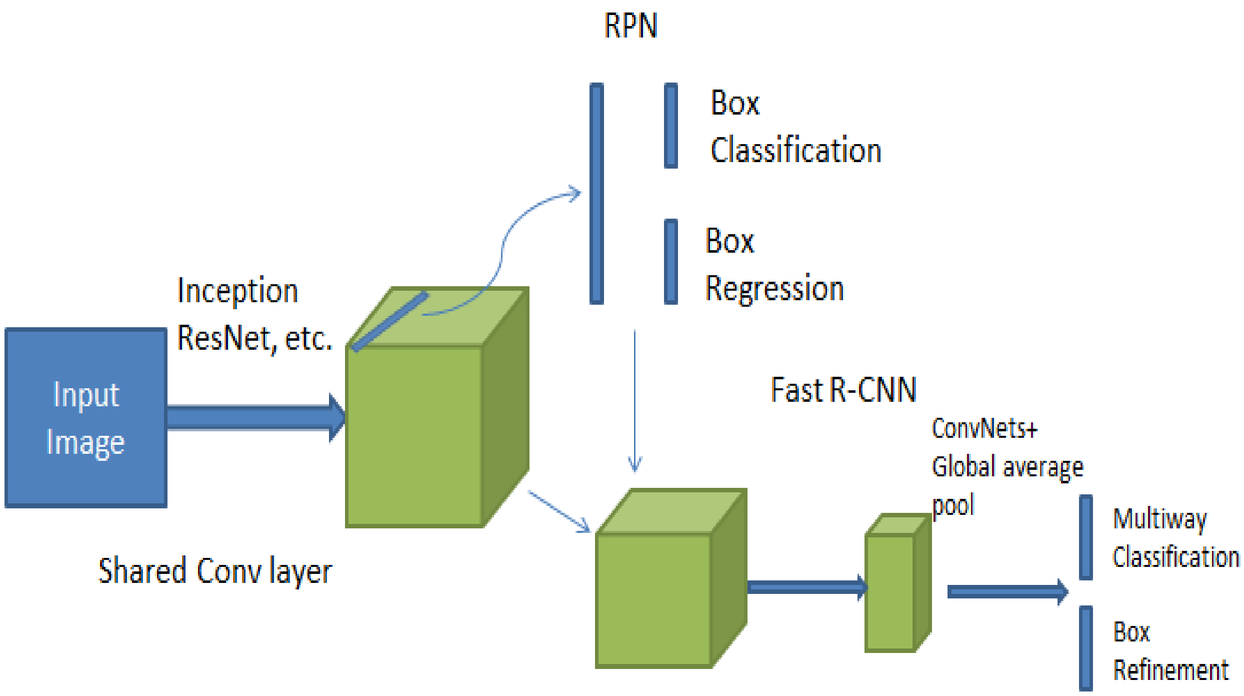
In the R-CNN architecture input image and multiple region of interest are the input in fully convolutional network. By the fully connected layers region of interest is pooled into a fixed size feature map and it is mapped with feature vector. In this network there are two input softmax probabilities and per-class bounding-box regression offset. It use to train end-to-end multi-task loss.

There is a region of interest is a rectangular window in a conv feature map. Faster R-CNN has two output layers. First is discrete probability distribution and second is the bounding-box regression offset. Faster R-CNN is the clean and fast update of R-CNN and SPnet.

2.6 Object Detection Based on Fast/Faster RCNN Employing Fully Convolutional Architecture

In this paper Yun Ren and Shunping Xaio from national university of defense technology have proposed the object detection based on Fast/Faster RCNN Employing Fully Convolutional Architecture.

There are two major parts which is feature extractor and feature classifier they can be considered as whole course in the modern object detectors. Feature extractor are usually hand engineered and feature classifier is usually nonlinear boost classifier. Fast RCNN and Faster RCNN makes further on the pipeline evolution in object detection. They make the best of use of softmax classifier and bounding-box regression rather than training softmax classifier. The system can take any image of any size as input it is proposed by ROI pooling layer.



In this architecture there are two main inputs which are region proposal network and second is the detection where fast RCNN is adopted. In this ROI pooling is used to pool region wise feature from the shared convolutional feature maps. By combining the network skip-layer connection is use in feature extraction. ResNet_50 is use due to lack of time to get detection in comparison with Fast RCNN. Last network stage is used to insert ROI pooling to ensure the effective output. But we set the output size of Rol pooling by 7*7. All the three layers are included in the last network stage.

III. CONCLUSION

The first paper worked only on a single form of compression (JPEG). Also, when looking for Blocking Artefacts, sometimes a higher textured region of an image may be mistaken for a manipulated region. This results in a lower accuracy of around 70%.

In the second paper, when only noise information in itself is used for image manipulation detection, the image has to retain the original noise information properly. Hence only uncompressed formats can be tested by this method. It gives an detection accuracy of 84%.

These methods generally aren't robust against compression, as compression can add its own noise/artefacts, altering or even ruining the original noise profile of the image. Also, they're not very effective against Gaussian smoothening applied on the image to mask its noise profile.

This problem can be reduced by R-CNN. This method divides the image into 2000 regions of interest (RoI) and feeds them to the CNN. But even here, 2000 regions is a huge amount, this makes it almost impossible to operate in real time.

When using Fast R-CNN, the number of inputs is reduced just to a single Image (Original Image) and the RoI is predicted by the same network. This reduces the computation further. But it is also observed that the responsibility of predicting the RoI slows it down significantly as compared to a scenario where the network isn't burdened with region proposals.

When using Faster R-CNN, the job to predict the RoI is outsourced to the Region Proposal Network (RPN) layer, which can be trained in itself. This makes it significantly faster. But it is also observed that a simple faster R-CNN RGB stream, can be insufficient against well-hidden contrast edges in the image, also it is not as effective in classifying the method of tampering which is better done by a noise stream. A noise stream in itself can't perform block regression as effectively as RGB network stream.

IV. ACKNOWLEDGMENT

We would like to thank our guide Mr. Deepak S. Shete for his guidance, support, motivation and encouragement in writing this review paper.

We would also like to convey our sincerest gratitude to the entire faculty members and staff of the Department of Electronics and Telecommunication Engineering, Thakur College of Engineering and Technology, who helped us when needed. A vote of gratitude to our fellow colleagues for their friendly co-operation and suggestions.

We would also like to show gratitude towards all the authors of the papers that are referred to in the process of authoring this particular paper

REFERENCES

[1] Dijana Tralic, Juraj Petrovic, Sonja Grgic 2012. JPEG Image Tampering Detection using Blocking Artifacts.

[2] Thibaut Julliand, Vincent Nozick and Hugues Talbot 2015. Automated Image Splicing Detection using RAW images, 6th International Conference on Imaging for Crime Prevention and Detection (ICDP-15).

[3] Dong-Hyun kim1, Hae-Yeoun lee 2017. Image Manipulation Detection using Convolutional Neural Network, International journal of applied Engineering Research ISSN.

[4] Belhassen Bayar, Matthew C. Stamm 2016. A Deep Learning Approach to Universal Image Manipulation Detection Using A New Convolutional Layer, International journal of IH & MMsec.

[5] Yen Ren and Shunping Xiao 2017. Object Detection Based on Fast/Faster RCNN Employing Fully Convolutional Architecture, University of defence technology, Changsha, China.

[6] Ross Girshick 2015. Fast R-CNN, Microsoft reaserch.