ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Issue 6, June 2016

IMAGE FORGERY DETECTION USING MEDIAN FILTERING BASED ON CONVOLUTIONAL NEURAL NETWORKS

A.GauthamRathina Kumar
PG Scholar
Department of IT
Noorul Islam University

Sugirtha
A/P Department of IT
Noorul Islam University

Abstract: The use of digital photography has increased over the past few years, a trend which opens the door for new and creative ways to forge images. The manipulation of images through forgery influences the perception an observer has of the depicted scene, potentially resulting in ill consequences if created with malicious intentions. This research explores the holes left by existing research; specifically, the ability to detect image forgeries created using multiple image sources and specialized methods tailored to the popular JPEG image format. In an effort to meet these goals, this paper presents four methods to detect image tampering based on fundamental image attributes common to any forgery. These include discrepancies in lighting, brightness levels, underlying edge inconsistencies and anomalies in JPEG compression blocks. Overall, these methods proved encouraging in detecting image forgeries with an observed accuracy of 95% in a completely blind experiment containing a mixture of authentic and forged images.

Keywords: Image forging, Digital Watermarking, Image Tampering, Median Filtering, Convolutional Neural Networks.

1. INTRODUCTION

Now a day's several software's are available that are used to manipulate image so that the image is look like as original. Images are used as authenticated proof for any crime and if this image does not remain genuine that it will create a problem. Detecting these types of forgeries has become serious problems at present. To determine whether a digital image is original or doctored is a big challenge. Three types of digital forgeries are popular. They are Image Retouching, Image splicing or photomontage and Copy –Move attack. Image Retouching is considered as less harmful kind of digital image forgery than other types present. In case of image retouching original image does not significantly changes, but there is enhancement or reduces certain feature of original image.Image splicing technique for making forgery images is more aggressive than image retouching. Image splicing is fundamentally simple process and can be done as crops and pastes regions from the same or separate sources.Copy-Move attack is more or less similar to image splicing in view of the fact that bot techniques modify certain image region(of a base image), with another image. However, instead of having an external image as the source, copy-move attack uses the portion of the original base image as its source. Active and Passive techniques can be used for image forgery detection.In active approach, the digital image requires some kind of pre-processing such as watermark embedded or signatures are generated at the time of creating the image. Passive image forensics is usually a great challenge in image processing techniques. There is not a particular method that can treat all these cases, but many methods each can detect a special forgery in its own way. The stream of passive tampering detection deals with analysing the raw image based

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Issue 6, June 2016

on various statistics and semantics of image content to localize tampering of image.Deep learning (deep machine learning, or deep structure learning, or hierarchal learning, or sometimes DL) is a branch of machine learning based on a set of algorithms that attempt to model high-level abstractions in data by using multiple processing layers with complex structures or otherwise, composed of multiple non-linear transformations.

2. DEEP LEARNING NEURAL NETWORKS

Deep learning is part of a broader family of machine learning methods based on learning representations of data. An observation (e.g., an image) can be represented in many ways such as a vector of intensity values per pixel, or in a more abstract way as a set of edges, regions of particular shape, etc.. Some representations make it easier to learn tasks (e.g., face recognition or facial expression recognition) from examples. One of the promises of deep learning is replacing handcrafted features with efficient algorithms for unsupervised or semi-supervised feature learning and hierarchical feature extraction.

A deep neural network (DNN) is an artificial neural network (ANN) with multiple hidden layers of units between the input and output layers. Similar to shallow ANNs, DNNs can model complex non-linear relationships. DNN architectures, e.g., for object detection and parsing generate compositional models where the object is expressed as layered composition of image primitives. The extra layers enable composition of features from lower layers, giving the potential of modelling complex data with fewer units than a similarly performing shallow network. DNNs are typically designed as feed-forward networks, but recent research has successfully applied the deep learning architecture to recurrent neural network for applications such as language modelling. Convolutional deep neural networks (CNNs) are used in computer vision where their success is well-documented.

3. PROPOSED SYSTEM

Whileanalyzing the proposed methods to detect copy-move images, the exact and robust matching techniques were found to be very promising. The suspect areas that have been determined to be matching are colored white, while all other areas are colored black. It appears that an image manipulation tool, such as “smudge”, has been used to try to hide the manipulator’s tracks. As stated before, this technique would only be good for image formats that do not use randomized lossy compression. The popular and widely used JPEG format would fail when using the exact matching technique. Therefore it needs some modification and it is done in the conventional CNN model by adding filter layer, and designs a specific CNN-based framework for median filtering forensics. Extensive experiments have shown that the proposed method can achieve better detection performance than the state-of-the-art schemes.

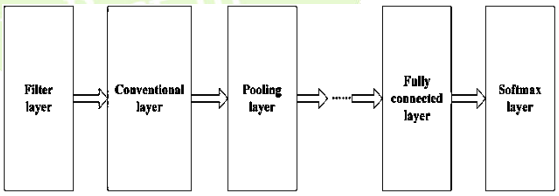

Fig.1 Proposed CNN model for Median Filtering Forensics

CNNs automatically learn features and perform the classification. It has deep architectures that consist of multiple levels of non-linear operations. A typical CNN has several types of layers, such as

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Issue 6, June 2016

convolutional layers, pooling layers and classification layers. At each convolutional layer, the output feature-map usually combines convolutions with multiple inputs. They can capture local dependencies among neighbor elements. The convolutional outputs from all inputs are then transferred into element-wise non-linearity. Pooling can reduce these partial resolution of each feature-map and translates information into more global one. Reported that max pooling can lead to faster convergence and improved generalization while analyzed the theoretical aspect of feature pooling. Via alternating convolutional layer and pooling

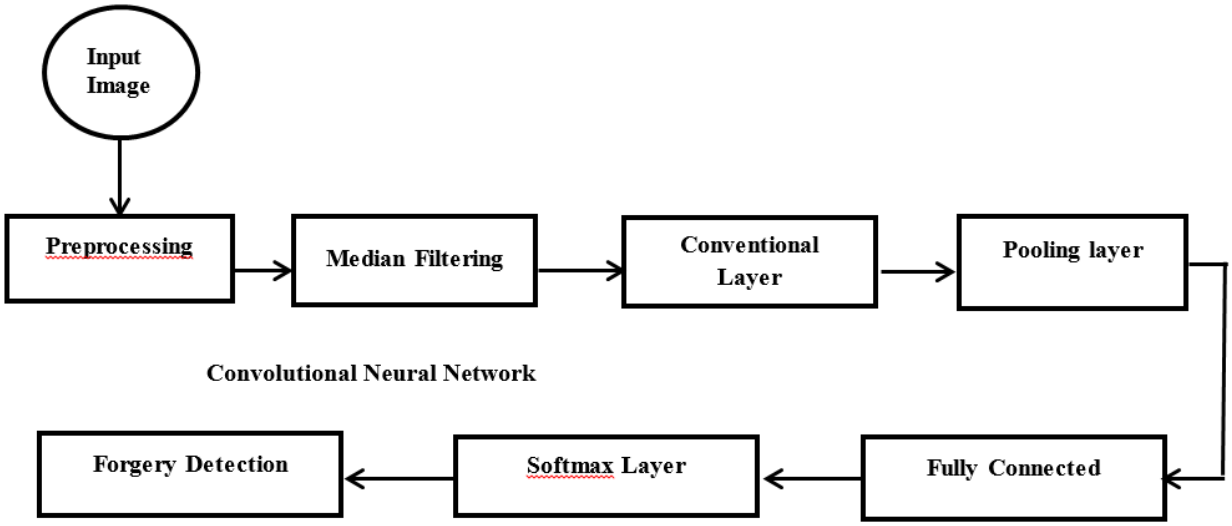

Fig. 2 Block Diagram of proposed CNN

layer, the output feature vectors are fed into the classification layer. Finally, the classification layer will output the probability of one sample classified into each class through softmax connection. In the preliminary study, it is directly employed to conventional CNN models as median filtering forensic models, and they didn't perform well, suggesting that existing CNN models can hardly capture the important statistical forensic properties. In median filtering forensics, since the fingerprint caused by median filtering is heavily affected by image edges and textures, using conventional CNN models directly (i.e., using the raw image pixels as inputs to CNNs) leads to poor performance. Therefore the propose model contains a modifying in the conventional CNN model by adding a filter layer due to the following intuitive reason: The added filter layer can suppress the interference caused by the presence of image edges and textures, and therefore the trace left by median filtering can be successfully exposed.

3.1 FILTER LAYER

Since in median filtering forensics, using conventional CNN models with the raw image pixels as inputs didn't yield good performances, one additional layer, the filter layer in Figure 4.1, is added to the conventional model. Through this

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Issue 6, June 2016

filter layer, the median filtering residual (MFR) of an image is obtained. Then the output MFR is fed into conventional networks. The filter layer is important in the proposed method since it can suppress the interference caused by image edges and textures. With eliminating/suppressing the interference of irrelevant information (e.g., image edges and textures), the trace left by median filtering can be investigated. The MFR can be expressed as,

$$d(i,j)=med_w(x(i,j)-x(i,j)) \text{ ..... (1)}$$

3.2 CONVOLUTIONAL LAYER

A conventional convolutional layer consists of two operations: convolution and non-linearity. The response of a convolutional layer is called feature map. Actually, each feature map is a particular feature representation of the input in a certain area. The convolution operation can be denoted as

$$x^l_j=\sum_{i=1}^{n} x^{l-1}_i * w^{l-1}_{ij} + b^l_j \text{ .....(2)}$$

The Rectified Linear Units (ReLUs) is used in our work because it can lead to fast convergence in the performance of large models trained on large datasets. Based on equation (2), the operation is expressed as

$$f_{m,n}=max(x^l_{m,n},0)\text{.....(3)}$$

3.3 POOLING LAYER

After obtaining feature maps using convolution, all the extracted features can be used for classification. However this can be computationally challenging and prone to over fitting. Thus only the mean (or max) value of a particular feature over a region of the image is calculated. The aggregation operation is called pooling. Average pooling and max pooling are two typical pooling methods, which propagate the average and the maximum value within the local region to the next layer respectively. The loss of spatial information is translated to an increasing number of higher level feature representations. The operations done in the pooling layer are shown in the Figure 4.4.

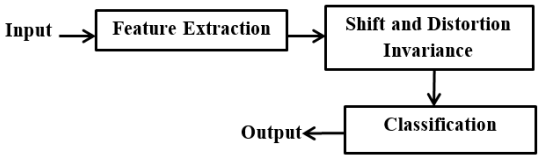

Fig. 4.4 Pooling layer

3.4 CLASSIFICATION LAYER

In general, the classification layer consists of a few fully connected layers. When the learned features pass through the first or two fully connected layers, they will be fed to the top layer of the CNNs, where a softmax activation function is used for classification. The back propagation algorithm is used to train the CNN. The weights and the bias can be renewed adaptively in the convolutional and fully connected layers following the error propagation procedure. In this way, the classification result can be fed back to guide the feature extraction automatically and the learning mechanism can be established.

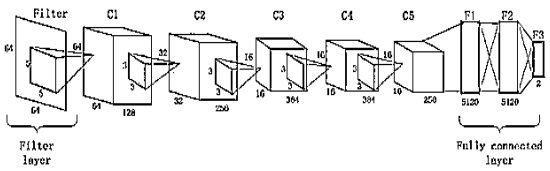

Fig 4.5 The framework of the proposed method.

In this work, it is important to address the challenge of detecting median filtering from a

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Issue 6, June 2016

small-sized and compressed image block. It is to be considering as two sizes of an input image, i.e. two different pixels of an image (64x64 or 32x32). Let it takes a gray scale image of size as the input to the architecture shown in Figure 4.3. Firstly, the filter layer gets the MFR of an image. Then the first convolutional layer convolves them with 128 kernels of size. The size of the output (C1) is, which means the number of feature maps is 128 and the resolution of feature maps is "64x64 ". Then the second convolutional layer takes the output of the first layer(C1) as the input and filters it with 256 kernels of size .The third, fourth, and fifth convolution layers apply convolutions with 384 kernels of size , 384 kernels of size ,256 kernels of size respectively. The Rectified Linear Units (ReLUs) is applied to the output of every convolutional layer. Meanwhile, the first, second, and fifth convolutional layers are followed by an overlapping max pooling operation with window size and step size 2, which operate on each feature map in the corresponding convolutional layer, and lead to the same number of feature maps with the decreasing spatial resolution. Each of the fully-connected layers (F1 and F2 in Figure 4.4) has 5120 neurons. In both fully-connected layers (F1 and F2), a recently-introduced technique, i.e., "dropout", is used. The last fully connected layer (F3) has two neurons. Its output is fed to a two-way softmax.

Fig. 4.6 Classification Block

3.5 SOFTMAX LAYER

The softmax function, or normalized exponential, is a generalization of the logistic function that "squashes" a *K*-dimensional vector $\mathbf{z}$ of arbitrary real values to a *K*-dimensional vector $\sigma(z)$ of real values in the range (0, 1) that add up to 1. The function is given by

$$\sigma(z)_j = \frac{e^{zj}}{\sum_{k=1}^{K} e^{zk}} \text{ for } j = 1,2 \ldots K \qquad (4)$$

The softmax function is the gradient-log-normalizer of the categorical probability distribution. For this reason, the softmax function is used in various probabilistic multiclass classification methods including multinomial logistic regression, multiclass linear discriminant analysis, naive Bayes classifiers and artificial neural networks. Specifically, in multinomial logistic regression and linear discriminant analysis, the input to the function is the result of *K* distinct linear functions, and the predicted probability for the *j*'th class given a sample vector **x** is:

$$P(y = j \mid x) = \frac{e^{x^T w_j}}{\sum_{k=1}^{K} e^{x^T w_j}} \qquad (5)$$

This can be seen as the composition of *K* linear functions $\mathbf{x} \mapsto \mathbf{x}^\mathsf{T}\mathbf{w}_1, \ldots, \mathbf{x} \mapsto \mathbf{x}^\mathsf{T}\mathbf{w}_K$ and the softmax function (where $\mathbf{x}^\mathsf{T}\mathbf{w}$ denotes the inner product of $\mathbf{x}$ and $\mathbf{w}$).

3.5.1 SOFTMAX IN ARTIFICIAL NEURAL NETWORK

In neural network simulations, the softmax function is often implemented at the final layer of a

ISSN 2395-695X (Print)
ISSN 2395-695X (Online)
International Journal of Advanced Research in Biology Engineering Science and Technology (IJARBEST)
Vol. 2, Issue 6, June 2016

network used for classification. Such networks are then trained under a log loss (or cross-entropy) regime, giving a non-linear variant of multinomial logistic regression.

Since the function maps a vector and a specific index *i* to a real value, the derivative needs to take the index into account:

$$\frac{\partial}{\partial q_k}\sigma(q,i) = \sigma(q,i)\left(\delta_{ik} - \sigma(q,k)\right) \quad (6)$$

Here, the Kronecker delta is used for simplicity (cf. the derivative of a sigmoid function, being expressed via the function itself).

3.5.2 SOFTMAX NORMALIZATION

Sigmoidal or Softmax normalization is a way of reducing the influence of extreme values or outliers in the data without removing them from the dataset. It is useful given outlier data, which includes in the dataset while still preserving the significance of data within a standard deviation of the mean. The data are nonlinearly transformed using one of the sigmoidal functions.

- The logistic sigmoid function:

$$x_i' \equiv \frac{1}{1 + e^{-\left(\frac{x_i - u_i}{\sigma_i}\right)}} \quad (7)$$

- The hyperbolic tangent function, *tanh*

$$x_i' \equiv \frac{1 - e^{-\left(\frac{x_i - u_i}{\sigma_i}\right)}}{1 + e^{-\left(\frac{x_i - u_i}{\sigma_i}\right)}} \quad (8)$$

The sigmoid function limits the range of the normalized data to values between 0 and 1. The sigmoid function is almost linear near the mean and has smooth nonlinearity at both extremes, ensuring that all data points are within a limited range. This maintains the resolution of most values within a standard deviation of the mean.

The hyperbolic tangent function tanh limits the range of the normalized data to values between -1 and 1. The hyperbolic tangent function is almost linear near the mean, but has a slope of half that of the sigmoid function. Like sigmoid, it as smooth, monotonic nonlinearity at both extremes. Also, like the sigmoid function, it remains differentiable everywhere and the sign of the derivative (slope) is unaffected by the normalization. This ensures that optimization and numerical integration algorithms can continue to rely on the derivative to estimate changes to the output (normalized value) that will be produced by changes to the input in the region near any linearization point.

4. CONCLUSION

Different from existing conventional median filtering forensics techniques, the feature extraction and classification steps are unified in a modified convolutional neural network (CNN) based model with adding a filter layer, and a hierarchical feature representations are learned. Using the feature representations learned automatically from a deep learning model, it can be achieved a better detection accuracy results when compared with the state-of-art methods using handcrafted features. It has been demonstrated that the proposed CNN based method can detect median filtering in small and JPEG compressed image blocks and is able to identify cut and paste forgeries well.Overall, this

method proved encouraging in detecting image forgeries with an observed accuracy of 95%.

Reference

[1] He T., Stankovic J.A., Lu C., and Abdelzaher T.(2003), 'Speed: A Stateless Protocol for Real-Time Communication in Sensor Networks,' in Proc. ICDCS, Vol.120,No.16 pp. 46-55.

[2] Bruckner D.,Picus C., Velik R., Herzner W., and Zucker G.(2012), 'Hierarchical semantic processing architecture for smart sensors in surveillance networks, IEEE Trans. Ind. Informat., Vol. 8, No. 2, pp. 291–301.

[3] Cheng L., Niu J., Cao J., Das S., and Gu Y.(2014), 'Qos aware geographic opportunistic routing in wireless sensor networks,' IEEE Trans. Parallel Distrib. Syst., Vol. 25, No. 7, pp. 1864–18

[4] LoBello L. and Toscano E.(2009), 'An adaptive approach to topology management in large and dense real-time wireless sensor networks,' IEEE Trans.Ind. Informat., Vol. 5, No. 3, pp. 314–324.

[5] Mao X., Tang S., Xu.X, Li X., and Ma H.(2011), 'Energy efficient opportunistic routing in wireless sensor networks,' IEEE Trans. Parallel Distrib. Syst.,Vol. 22, No. 11, pp. 1934–1942.

[6] Toscano E. and Lo Bello L.(2008), 'A topology management protocol with bounded delay for wireless sensor networks,' in Proc. IEEE Int. Conf. Emerging Technol. Factory Autom., ETFA 2008, , pp.942–951.

[7] Lu C., Blum B., Abdelzaher T., Stankovic J., and He T.(2002), 'RAP: A Real-Time Communication Architecture for Large-Scale Wireless Sensor Networks,' in Proc. IEEE RTAS, pp. 55-66.

[8] Juan Luo, Jinyu Hu, Di Wu, and Renfa Li(2015), 'Opportunistic Routing Algorithm For Relay Node in Wireless Sensor Networks', IEEE Transactions On Industrial Informatics,Vol. 11, No. 1, pp. 112-121.

[9] Mao X., Tang S., Xu.X, Li X., and Ma H.(2011), 'Energy efficient opportunistic routing in wireless sensor networks,' IEEE Trans. Parallel Distrib. Syst.,Vol. 22, No. 11, pp. 1934–1942.

[10] Toscano E. and Lo Bello L.(2008), 'A topology management protocol with bounded delay for wireless sensor networks,' in Proc. IEEE Int. Conf. Emerging Technol. Factory Autom., ETFA 2008, , pp.942–951.

[11] Ren F., Zhang J., He T., Lin C., and Ren S.K.(2011),'EBRP: Energybalanced routing protocol for data gathering in wireless sensor networks,' IEEE Trans. Parallel Distrib. Syst., Vol. 22, No. 12, pp. 2108–2125.

[12] Younis O. and Fahmy S.(2004), 'HEED: A Hybrid, Energy-Efficient Distributed Clustering Approach for Ad Hoc Sensor Networks,' IEEE Trans. Mobile Computing, Vol. 3, No. 4, pp. 366-379.

[13] Xu Y., Heidemann J., and Estrin D.(2001), 'Geography-Informed Energy conservation for Ad Hoc Routing,' Proc. Seventh Ann. Int'l Conf. Mobile Computing and Networking.

[14] Zhang D., Li G., Zheng K., Ming X., and Pan Z.H.(2014), 'An energy-balanced routing method based on forward-aware factor for wireless sensor network,' IEEE Trans. Ind. Informat., Vol. 10, No. 1, pp. 766–773.