Abstract

The technology today is well developed that images can be easily manipulated and tampered with various digital tools that we can no longer rely on these images. Doctored images have become ubiquitous and they are so real that they leave very little evidence of being tampered. There have been many attempts to detect such doctored images. Recent proposals have used linear models to represent interpolations in color array filters in digital cameras to identify the forged images. In practice, a nonlinear interpolation is done to perform this task. In this paper, we propose the idea of using Back propagation Neural network as the nonlinear model to represent this interpolation. The features collected from the model were subjected to dimensionality reduction using K-LDA to formulate the NN, NM and SVM classifier and a reasonable success rate of 61.2 % was obtained in identifying the forged image.

Keywords: Color array filter, Back propagation Neural Network, kernel LDA, feature vectors, forged image

1. Introduction

* Corresponding author. Tel.: +91 9445754818.
E-mail address: sankarramanvinoth@gmail.com (Vinoth S)
With the availability of tools like Photoshop, paint etc. forging images have become very easy. Such forged images are made by rotation, rescaling, copy-move, image slicing, stretching, zooming, geometric transformation etc. Image forgery detection has become very important especially in situations where digital images are produced as evidence and so these are necessary to ensure the originality of such images. Currently available forgery detectors are classified as passive and active. Active detector uses data hiding concepts like water marking to identify the forged part. On the other hand, passive detector only uses the raw image for detecting the forgery. In this paper a passive forgery detector for color image is proposed.

Color images from digital cameras usually need 3 sensors to sense red, green and blue contribution for every pixel of the image. Color filters are needed because typical photo sensors detect light with very little wavelength specificity and therefore cannot separate color information. The most commonly used color filters are Bayer filters.

![Color filter of a 3×3 block (Left), G- values taken directly from the sensors and g- interpolated values (Right)](image)

Here each pixel senses only one of the three colors and the remaining two colors are estimated from the values of the nearby pixels. As shown in the figure, the pixel at (2, 2) (Refer Figure 1 (Left)) measures only green. The blue content of this pixel is estimated from the color information from the neighboring pixel which measure blue i.e. (2, 1) and (2, 3). Likewise the red content is estimated from the sensors placed at (1, 2) and (3, 2). These values are interpolated from the nearby pixel values as explained above and the way these values are interpolated differs with each camera.

There are many proposals currently available to do this interpolation by a demosaicing algorithm which is tailored for each type of color filter. Here the unknown color information is having a nonlinear relationship (based on the type of demosaicing algorithm used) with the known corresponding color information of the nearby surrounding pixel values. For example, in Fig 1 the unknown green values g6, g7, g8, g9 of the 3×3 sized green channel are related nonlinearly with the green values G1, G2, G3, G4, G5 (obtained using sensors). In this paper the nonlinear relationship using Back propagation Neural network (BPNN) is proposed to model this relationship as described in section 2.

2. Proposed nonlinear relationship using Back propagation Neural Network model

Mathematically the proposed 1×4 sized output vector \( O = [g_1, g_2, g_3, g_4] \) is related with 1×5 sized input vector \( I = [G_1, G_2, G_3, G_4, G_5] \).
[G₁G₂G₃G₄G₅] (which are obtained from the arbitrary 3×3 sized green channel) (Refer Figure 1 (Right)) in the following manner. \( H = f₁(I \times W₁ + B₁) \) and \( O = H \times W₂ + B₂ \) where \( H \) is 1×2 sized hidden vector, \( W₁ \) is 5×2 sized weight matrix (between input layer and hidden layer), \( B₁ \) is 1×2 sized biased vector at hidden layer, \( W₂ \) is 2×4 sized weight matrix (between hidden layer and output layer), \( B₂ \) is 1×4 sized bias vector at the output layer. \( f₁ \) is chosen as the nonlinear function \( \text{logsig}(n) = \frac{1}{1+\exp(-n)} \). This relation is illustrated in the Fig 2.

Suppose the digital image captured using the digital camera is subjected to manipulation like rotation, resize, scaling etc. using the image edit tool boxes like photo shop and paint the relationship above gets distorted and is no more valid. The values of the above mentioned matrices and vectors (\( W₁, B₁, W₂ \) and \( B₂ \)) are arranged in a single row to obtain a feature vector of size 1×24. Thus the feature vector of size 1×24 obtained as described helps in identifying whether the particular part of the image is original or forged. The proposed feature vector is \([W₁ \text{ (stacked in a single row)} \ W₂ \text{ (stacked in a single row)} \ B₁ \text{ (stacked in a single row)} \ B₂ \text{ (stacked in a single row)}]\)

![Fig. 2. Proposed Neural network model. Refer Fig 1 for G1, G2, G3, G4, G5 and g1, g2, g3, g4, g5](image)

3. Proposed forgery detector

3.1. Feature vector extraction

A total of 100 images captured from 9 different digital cameras (Canon EOS REBEL T3i, NIKON D5100, NIKON D3100, MID ST8, SONY DSC-W210, NIKON COOLPIX P100, SONY ERICSSON SK17i, SAMSUNG GT S6102, CANONPowerShot A590 IS) were collected. From each of the image, 500 3×3 sub blocks were collected at random as shown in Fig 3. 1×5 sized input vectors and the corresponding 1×4 sized output vectors were collected from every 3×3 sub blocks of the green channel matrix. In a Bayer array filter the number of green sensors is more than the number of red and blue sensors. So, in this paper we have used the green values taken from the image to model the neural network.
The feature vectors are obtained as mentioned in section 2. This vector completely describes the interpolation relationship.

The input and output vectors collected from these 500 sub blocks were used to train the BPN to get one feature vector of size 1×24. This is repeated for all the input output vectors from each image and this formed the feature vectors for the original images. Doctored images were created by copy paste, resize, rotation etc. and the input and output vectors were collected from the doctored part of the image and the above steps were repeated to extract feature vectors for the forged images.

To check the accuracy with which the neural network has modeled the camera 50 such sub blocks were taken from the same image and the predicted outputs and the corresponding original values were plotted. Expected values are the values taken from the corresponding pixel of each image (interpolation done by the camera) and predicted values are those done by the trained neural network. Fig 4 shows such a plot for one such sample image. A sample of the forged image we used is shown in Fig 5.
Fig. 5. (a) Sample original images

Fig. 5. (b) Sample forged image. Note: The forged image was created with the two sample images shown in Fig. 5 (a). The white colored cat shown in the left side image of Fig 5(a) has been merged with the right side image.
3.2. Dimensionality reduction, training the classifier and testing phase

About 100 feature vectors were collected from both original and forged images. 50 such feature vectors (collected from 4 different cameras in case they were from original data set) of them were used for training. They were projected onto kernel space (7 different kernels were used) and LDA was performed to reduce their dimension from $1 \times 24$ to 1. This one dimensional data was used to train our classifier. We used three different types of classifier: SVM, nearest neighbor and nearest mean. Five different kernels were used for SVM. The remaining 50 feature vectors were used for testing. Their dimensions were reduced from $1 \times 24$ to 1 as in the testing phase and were fed as inputs to the three different classifiers. The outputs are tabulated in Table 1. The proposed technique is explained in Fig 6.

Table 1. Percentage of success

<table>
<thead>
<tr>
<th>Kernel for LDA</th>
<th>Kernel for Classifier</th>
<th>SVM with linear kernel</th>
<th>SVM with quadratic kernel</th>
<th>SVM with polynomial kernel</th>
<th>SVM with rbf kernel</th>
<th>SVM with mlp kernel</th>
<th>Nearest mean</th>
<th>Nearest neighbour</th>
</tr>
</thead>
<tbody>
<tr>
<td>Innerproduct kernel</td>
<td>SVM</td>
<td>45.918367</td>
<td>56.122449</td>
<td>55.102041</td>
<td>38.77551</td>
<td>41.836735</td>
<td>45.918367</td>
<td>43.877551</td>
</tr>
<tr>
<td>Gaussian</td>
<td>SVM</td>
<td>45.918367</td>
<td>56.122449</td>
<td>55.102041</td>
<td>38.77551</td>
<td>41.836735</td>
<td>45.918367</td>
<td>43.877551</td>
</tr>
<tr>
<td>Polynomial</td>
<td>SVM</td>
<td>53.061224</td>
<td>54.081633</td>
<td>53.061224</td>
<td>53.061224</td>
<td>48.979592</td>
<td>53.061224</td>
<td>54.081633</td>
</tr>
<tr>
<td>Power exponential</td>
<td>SVM</td>
<td>45.918367</td>
<td>58.163265</td>
<td>54.081633</td>
<td>39.795918</td>
<td>43.877551</td>
<td>45.918367</td>
<td>42.857143</td>
</tr>
<tr>
<td>Hyperbolic tangent</td>
<td>SVM</td>
<td>48.979592</td>
<td>48.979592</td>
<td>50</td>
<td>50</td>
<td>51.020408</td>
<td>48.979592</td>
<td>52.040816</td>
</tr>
<tr>
<td>Cauchy</td>
<td>SVM</td>
<td>53.061224</td>
<td>56.122449</td>
<td>55.102041</td>
<td>38.77551</td>
<td>41.836735</td>
<td>45.918367</td>
<td>46.938776</td>
</tr>
<tr>
<td>Inverse multi-quadric</td>
<td>SVM</td>
<td>45.918367</td>
<td>56.122449</td>
<td>55.102041</td>
<td>38.77551</td>
<td>41.836735</td>
<td>45.918367</td>
<td>41.836735</td>
</tr>
</tbody>
</table>

![Fig. 6. Block diagram of the proposed technique](image-url)
4. Conclusion and future work

Out of the 100 feature vectors 50 were used to train the classifier and the remaining 50 were used for testing. Care was taken to consider features from 4 different cameras for training and 5 different cameras were used for testing. The nonlinear model proposed is experimentally shown to have worked well with a reasonable detection of 61.2%. We have used back propagation neural network and kernel LDA for getting the features. Other methods like LPC, ICA, and IDA could well be employed in place of kernel LDA to get better classification and other models of neural network could be used to achieve better classification.

References