

## Article

# HSB-SPAM: An Efficient Image Filtering Detection Technique

 Saurabh Agarwal <sup>1,2</sup> and Ki-Hyun Jung <sup>2,\*</sup> 
<sup>1</sup> Amity School of Engineering & Technology, Amity University Uttar Pradesh, Noida 201301, India; saurabhnsit2510@gmail.com

<sup>2</sup> Department of Cyber Security, Kyungil University, Gyeongsbuk 38428, Korea

\* Correspondence: khannyjung@gmail.com; Tel.: +82-53-600-5626

**Abstract:** Median filtering is being used extensively for image enhancement and anti-forensics. It is also being used to disguise the traces of image processing operations such as JPEG compression and image resampling when utilized in image de-noising and smoothing tool. In this paper, a robust image forensic technique namely HSB-SPAM is proposed to assist in median filtering detection. The proposed technique considers the higher significant bit-plane (HSB) of the image to highlight the statistical changes efficiently. Further, multiple difference arrays along with the first order pixel difference is used to separate the pixel difference, and Laplacian pixel difference is applied to extract a robust feature set. To compact the size of feature vectors, the operation of thresholding on the difference arrays is also utilized. As a result, the proposed detector is able to detect median, mean and Gaussian filtering operations with higher accuracy than the existing detectors. In the experimental results, the performance of the proposed detector is validated on the small size and post JPEG compressed images, where it is shown that the proposed method outperforms the state of art detectors in the most of the cases.

**Keywords:** image forensics; higher significant bit-plane; median filtering detection; image forgery detection; fake image; Markov chain



**Citation:** Agarwal, S.; Jung, K.-H. HSB-SPAM: An Efficient Image Filtering Detection Technique. *Appl. Sci.* **2021**, *11*, 3749. <https://doi.org/10.3390/app11093749>

Academic Editor: Ugo Vaccaro

Received: 23 March 2021

Accepted: 16 April 2021

Published: 21 April 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Today, most of the communication is being done in digital form, which opens a Pandora box of security attacks such as editing, copying, deleting and so on. Sometimes, the editing operation which does not require any expertise due to the wide availability of ubiquitous digital imaging devices and sophisticated editing software, may create an uproar as fake (edited) images and can be circulated through popular social media platforms to mislead the users as shown in Figure 1. Figure 1a shows that the anti-gun activist and student Emma Gonzalez is tearing the copy of US constitution.



(a) Fake picture

(b) Original picture

**Figure 1.** Two pictures: (a) fake picture; (b) original picture.

Whereas, the real picture is shown on the right side published in Teen Vogue magazine that is tearing the gun range target photo. The fake image was circulated on social media more than 70,000 times to get some political mileage in 2018. Therefore, the need of forensic toolboxes is required strongly which can assess blindly the authenticity of digital images without access to the source image, source device or the aid of an auxiliary watermark signal.

In the recent years, image forgery detection techniques [1–3] based on handcrafted features and CNN features have been introduced. Matern et al. [1] proposed a physics-based method that considers the 2-D lighting environments of objects. The incident of the light assists in detecting fake objects in the image. The method is verified on high-resolution challenging databases and has proved its superiority for some existing techniques, however there is still sufficient gap for improvement. Moghaddasi et al. [2] exploited the discrete cosine transform (DCT) domain and singular value decomposition for image forgery detection. The method uses moments as features and support vector machine (SVM) as a classifier. The authors claim the good accuracy on multiple databases although performance may be degraded on cross-dataset and unknown image. Deep networks are found effective in many applications like face reorganization [4], object classification [5], facial expression [6] and so on. Marra et al. [3] proposed a CNN based feature extractor for high-resolution images. Unlike other CNN techniques, the gradient check-pointing is applied for feature extraction on entire original image size. Marra et al.'s technique extracts better features than other techniques in which the image is divided into overlapping patches to extract features. However, it has a high computational cost as full-size images are being used in feature extraction. In addition to the aforementioned handcrafted features and CNN features-based techniques, there has been discussed several image forgery detection techniques which are based on hardware or camera fingerprints, lighting environment, illumination, statistical information, deep networks and so on. In spite of this, some techniques [7–9] are also designed to counter image forensic and forgery detection techniques. Marra et al. [7] designed the technique to counter the image forgery detection techniques. Kim et al. [8] applied the adversarial Networks to remove the artifacts of median filtering. Gragnaniello et al. [9] discovered that CNN based techniques are easy to conquer than machine learning based image forgery detection techniques while fake images are generated by an adversarial network.

To make the fake images more realistic, the malicious intent people use multiple operations such as rotation, resampling, contrast enhancement so that the traces of modifications are hidden. Thus, the performance of image forgery detection methods is adversely affected by the filtering operations such as median filter, Gaussian filter, and mean filter. In particular, the median filtering (MF) is preferred among some forgers because it has the characteristics of nonlinear filtering based on order statistics. Therefore, a Markov chain based median filtering detector using a high significant bit-plane is proposed in this paper. The Markov chain is popular in many applications such as steganalysis, object classification, atmospheric science, and even in median filtering detection. The prominent median filter detector using Markov Chain was proposed by Kirchner and Fridrich [10] which uses difference arrays for Markov chain. Alike method is also proposed by Pevny et al. [11] for steganalysis and popularly known as SPAM. The method extracts features using Markov chain on thresholded difference array that results into more granular collection of features and ultimately leads to a higher accuracy. Chen et al. [12] considered the absolute difference while applying thresholding to minimize the size of feature set so that the complexity of the method can be reduced. Cao et al. [13] calculate the variance of difference arrays and use the probability of zero elements to decide whether an image is median filtered or not. Cao et al.'s method is close to Markov model. Agarwal and Chand [14] extended the work of Kirchner and Fridrich's method [10] by considering higher order differences. The incorporation of higher order differences collects even the minute details of images that help in effective collection of feature sets. The extraction of features is executed with the help of Markov chain on the first, second and third order

thresholded difference arrays. Thus, Agarwal and Chand's method could improve upon Kirchner and Fridrich's method. However, it has been observed that the thresholded array and absolute difference can be lose some relevant information which may be useful in more accurate detection. In this paper, a high significant bit-plane based image forgery detection strategy is proposed to retain vital information. The main contributions of the proposed method can be summarized as follows:

- The proposed scheme can reduce the bit depth to attain more relevant information in difference arrays. Higher significant bit-planes are considered for strong statistical analysis.
- Various derivatives such as pixel difference, separate pixel difference, and the Laplacian operator are considered for a robust feature vector that can help in collecting additional information so that accuracy can be further enhanced. Further, the co-occurrence statistics is extracted from derivatives using the Markov chain.
- Experimental results are compared with some of the popular methods such as GLF, GDCTF, PERB and SPAM to evaluate the performance of the proposed scheme. For the exhaustive analysis, the experimental results are shown for median filtered, mean filtered and Gaussian filtered images.
- The experimental analysis shows that the proposed method has better performance than existing methods in most of the scenarios. There is a significant improvement of more than 2% in detection accuracy for the case of  $3 \times 3$  size filter on the low-resolution images and highly compressed images.

The rest of the paper is organized as follows. In Section 2, existing techniques are discussed. The proposed technique is elaborated in Section 3. In Sections 4 and 5, experiment results and conclusion of the paper are discussed respectively.

## 2. Related Work

In this section, the existing and popular filtering detection techniques are briefly reviewed. It has been observed that there are many works [10,12–35] for detecting filtering in which most of works give emphasis on median filtering detection due to its non-linear nature. Only some methods in [10,23] exist for detecting Gaussian filtering and for median, mean and Gaussian filtering in [23,24,31]. The existing techniques can be categorized as the autoregressive model, the CNN model, frequency domain, and Markov chain.

Kang et al. [16] introduced the autoregressive model for median filtering detection. Median filter residuals are considered and three median filter residuals arrays are generated after applying median filter of different size. The size of feature vector is small though a performance degrades on low resolution images. Ferreira and Rocha [17] also generated numerous median filter residuals arrays after applying median filter multiple times. The eight quality matrices are calculated on residual arrays. The SVM classifier is applied to classify the images using quality matrices as a feature vector. The method fails on low resolution and compressed images. Yang et al. [21] proposed a two dimensional autoregressive (2D-AR) model. The 2D-AR model was applied on median filtered residual (MFR), average filtered residual (AFR) and Gaussian filtered residual (GFR). Multiple residuals are still considered the accuracy decreases on small sized images. Rhee [27] proposed a technique that utilizes the autoregressive model, edge detectors and Hu moments. The hybrid feature set is not able to tackle the compression issue. In addition, Rhee [28] also applied the autoregressive model on bit plane sliced images. Rhee applied the autoregressive model on bit plane sliced median filtered residual arrays. The least and highest significant bits are considered and the performance is verified on resampled images. Peng et al. [31] proposed a hybrid feature set using autoregressive model and local binary pattern (LBP) operator. The proposed model was able to classify median, mean and Gaussian filtered images. Chen et al. [18] introduced the convolutional neural network (CNN) for median filtering detection. The new CNN layer is introduced to provide median filter residual. The performance of method is superseded by Tang et al. [22] by proposing a CNN model using two multilayer perceptron (mlpconv) layers. The mlpconv layer is claimed more suitable

for nonlinear data learning. Bayar and Stamm [23] introduced a constrained convolutional layer to highlight the image features. The method can detect median filtering, Gaussian blurring, resampling and JPEG compression. Luo et al. [29] applied the high pass filter on the image and resultant image is given as input to CNN. Zhang et al. [35] proposed a CNN model by introducing an adaptive filtering layer (AFL). The proposed AFL is constructed to capture the frequency traces of median filtering in the DCT domain. The performance of CNN techniques is satisfactory on low resolution images although computation cost is very high and is required large dataset for training.

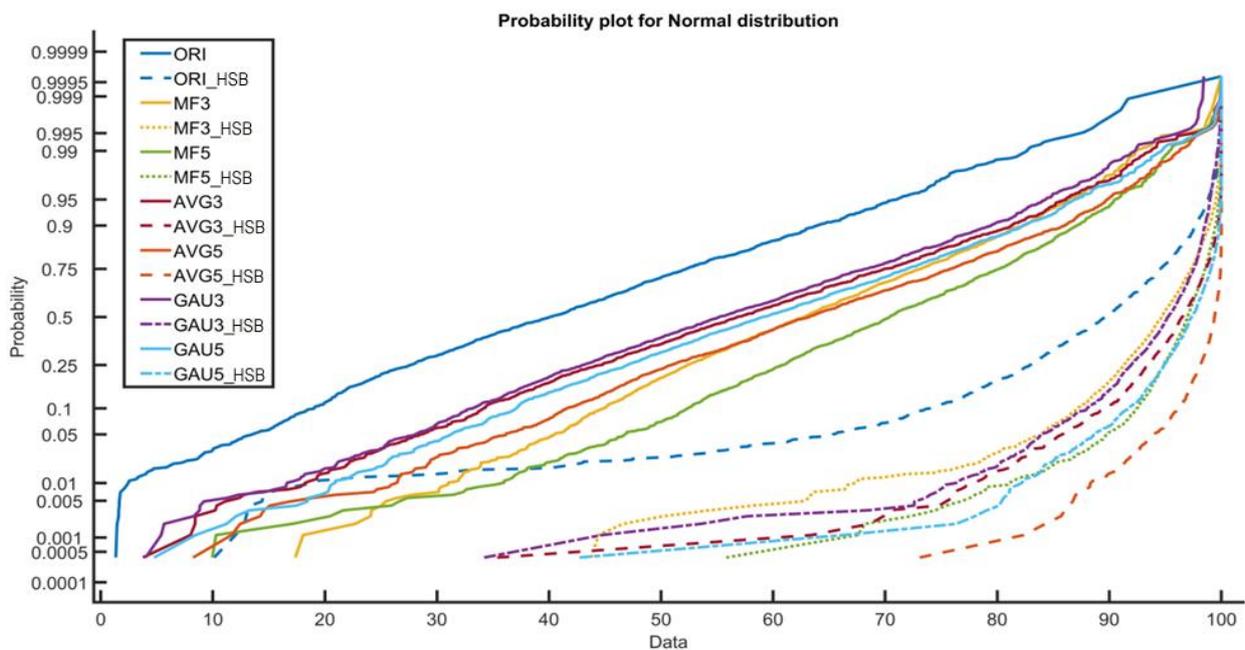
Hwang and Rhee [19] proposed a method for Gaussian filtering detection. The Gaussian filter residual and frequency transform residual features are utilized for detection. The fast Fourier transform is used to convert from the spatial domain to the frequency domain. Liu et al. [20] also considered the Fourier transform for the frequency domain. In Liu's method, it is discovered that most of the values are zero in a low-frequency region of the median filtered image in contrast to a non-filtered image. Wang and Gao [24] proposed a local quadruple pattern and extracted features in discrete cosine transform (DCT) to detect median, mean and Gaussian filtering. Li et al. [25] also proposed a method using the DCT domain. The addition of absolute differences of DCT coefficients are calculated on sequential median filtered images. The SVM classifier is used to classify the images. Gupta and Singhal [30] proposed a 4-dimension feature set using median filter residual in DCT domain. Hwang and Rhee [32] discussed a 10-dimension feature set for Gaussian filtering detection. The amount of blurring is calculated from Gaussian filtered images of ten different filter size. The frequency-based techniques are not giving the good results as the change in domain loose important statistical information. Kirchner and Fridrich [10] considered the difference arrays for median filtering detection. The ratio of difference arrays histogram bins is analyzed to discover uncompressed median filtered images. The authors also proposed a technique for a compressed median filtered images using Markov chain on difference arrays although detection accuracy is low on the small size images. Chen et al. [12] proposed a hybrid feature set using global and local features. Global features are extracted using the Markov chain and local features using normalized cross correlation on difference arrays. The absolute difference is considered while applying the Markov chain on difference arrays. The absolute difference loses the vital information change. Agarwal and Chand [14] applied the Markov chain on the first, second and third order difference arrays. There is performance gain by considering a higher order differences. Niu et al. [15] combined the features of uniform local binary pattern and Markov model for better results. Peng et al. [26] considers multiple residuals—first order difference, median filtering residual, and median filtering residual difference. The autoregressive model and the Markov chain are applied to fetch statistical information from multiple residuals. Gao and Gao [33] applied the autoregressive model in the frequency domain and the Markov chain in the spatial domain. The method applied on difference arrays in multiple directions. Gao et al. [34] combined two feature sets—Markov chain and local configuration patterns features, where the combined feature set approach gives slight performance improvement. In the most of the Markov chain based methods [10,12,13,15,26,33,34], the SVM classifier is utilized to classify into non-filtered and median filtered images, where the linear discriminant analysis classifier is applied for classification. Markov chain represents the co-occurrence of elements in a better way in comparison with the autoregressive model and the frequency domain based methods. There is sufficient loss of statistical information while converting from spatial to frequency domain. The autoregressive model gives a small size feature vector but there is a need of optimizable parameters for training of each scenario (JPEG compression, image size). Further, experimental analysis showed that the performance of many methods based on autoregressive and frequency domain is good only due to zero padding artifacts. The computation cost of CNN based methods is very high and also there is a need of large dataset for training. In this paper, Markov chain is utilized in spatial domain for its simplicity and robustness. The proposed work is discussed in the next section.

### 3. The Proposed Method

The proposed method consists of two phases namely feature extraction, and training and classification. The contribution of the proposed method lies in the feature extraction phase in major. For an elaborative study and analysis, the feature extraction phase is subdivided into a higher significant bit-plane analysis and pixel difference arrays and Markov chain sub-phases. In this section, a higher significant bit-plane analysis is described followed by a discussion on pixel difference arrays and the Markov chain.

#### 3.1. Higher Significant Bit-Plane Analysis

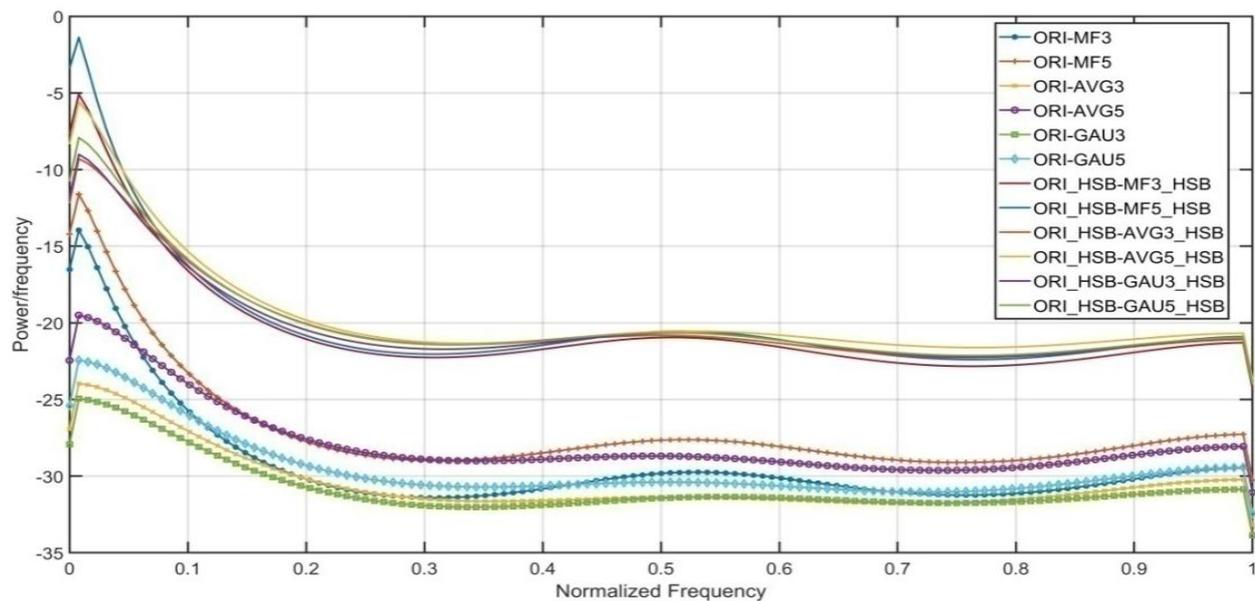
In order to collect more valuable statistical information, the proposed method transforms the image into a higher significant bit-plane based image by dropping least significant bits of each pixels and considering only the higher significant bits. The resultant image contains more relevant information in particular range that helps in generating more valuable information with difference arrays than the complete image. To study experimentally and analyze the effectiveness of the higher significant bit-plane based image, a normal probability distribution of pixel difference in the UCID database [36] having 1338 images is considered. The graph of the normal probability distribution is shown in Figure 2, where ORI denotes the non-filtered image, ORI\_HSB denotes the higher significant bit-plane based image of ORI images. In general, the 8-bit grayscale image is considered by conventional image filtering detection techniques. However, 4-HSB version of images is considered in this paper. Median filtered image with filter size  $3 \times 3$  and  $5 \times 5$  are denoted as MF3 and MF5 and their corresponding higher significant bit-plane versions are denoted by MF3\_HSB and MF5\_HSB. In the same way, mean and Gaussian filtered images with filter size  $3 \times 3$  &  $5 \times 5$  are denoted as AVG3, AVG5, GAU3 & GAU5, respectively. It is evident from the Figure 2 that the higher significant bit-plane filtered images (MF3\_HSB, MF5\_HSB, AVG3\_HSB, AVG5\_HSB, GAU3\_HSB, GAU5\_HSB) contain a most of the information in small interval whereas the information is distributed into a large interval in normal images. Since the Markov chain is applied on thresholded arrays, the higher significant bit-plane images will perform better in the view of thresholding.



**Figure 2.** Normal distribution of 8-bit and 4-bit images.

To further verify the benefit of higher significant bit-plane images, entropy of the first order pixel difference arrays is calculated from the non-filtered and filtered images. The

entropy dissimilarity of pair ORI and MF3 is less in comparison with pair ORI\_HSB and MF3\_HSB. The high dissimilarity of low-bit filtered and low-bit non-filtered images give better detection capability. The main reason is that difference array of higher significant bit-plane images performs better. Further, the dissimilarity pattern is also followed by MF5, AVG3, AVG5, GAU3 and GAU5 filtered images as shown in Figure 3.



**Figure 3.** Entropy distribution of non-filtered and filtered difference pairs.

As discussed above, the Markov chain is applied on thresholded difference arrays to extract statistical information in compact array, the higher significant bit-plane images provides better information for feature sets without increasing the size of feature vector. The validation of performance benefits in the higher significant bit-plane images is illustrated in Figure 4. In the example, a  $5 \times 5$  size pixels image patched from 8-bit image is considered as 8-bit image is commonly used for experimental analysis. The horizontal difference array of the image is further thresholded at threshold  $\Gamma = 3$ . To analyze the variation in information, the change array is constructed. The change array shows the change in the element value of column wise. Since the initial three values are  $-3$  in the first row of thresholded array, there is no change and it is represented in the change array as 0. The fourth value in the first row of thresholded array is three. It means that the value is changed from  $-3$  to 3. Therefore, the change is represented by one. The same analysis is performed on 4-HSB image of the same 8-bit image which is considered in the aforementioned case. It can be seen in the thresholded array and change array that shows better variation on 4-HSB image means that the statistical analysis performed on HSB image will demonstrate better filtering detection accuracy.

In addition to 4-HSB depth, other bit-depths can be also considered as a base on the image data set characteristics. However, it has been observed from the experimental analysis that selection of 4-HSB depth has been the most appropriate for high accuracy. In the next sub-section, the discussion regarding multiple pixel difference arrays is provided to extract robust statistical information, which is followed by the discussion of Markov chain to extract feature vector on the difference arrays.

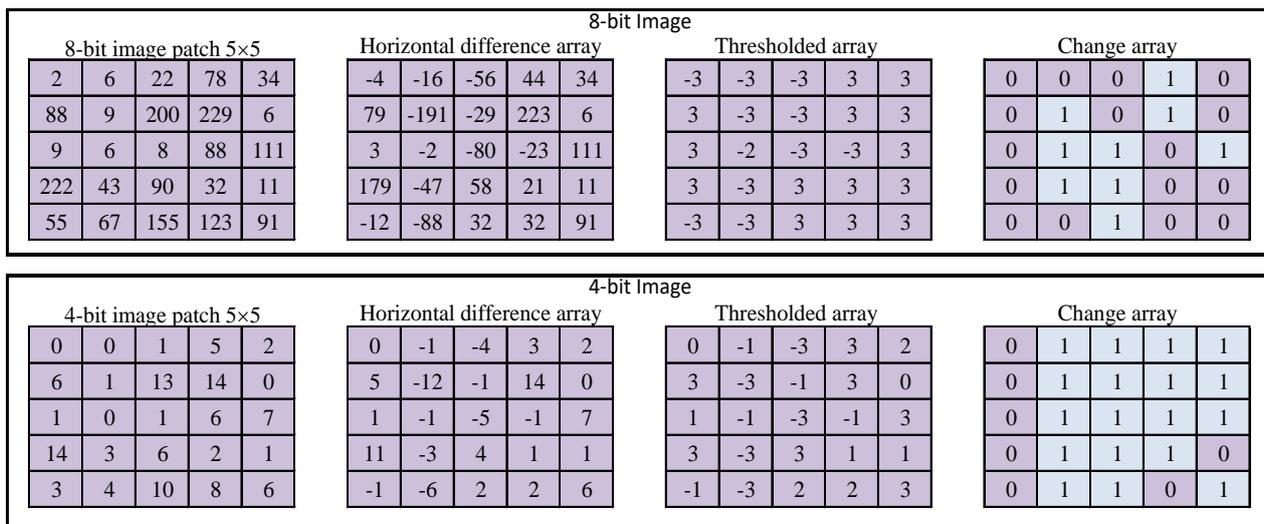


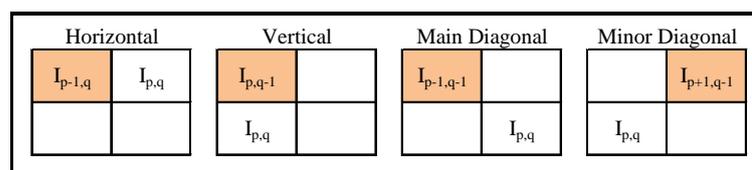
Figure 4. The 8-bit and 4-bit depth image statistical analysis.

### 3.2. Pixel Difference Arrays and Markov Chain

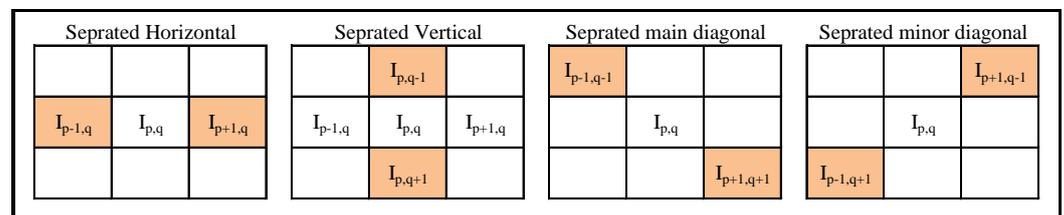
The first order pixel difference array and Markov chain has been used in literature to extract the feature set. The extracted set of features has been used to train the classifier that would be used for classification of images finally. The first order pixel difference array contains from range  $-255$  to  $255$  in the case of 8-bit image. The first order pixel difference array in the vertical direction can be represented as follows for an image  $I$ :

$$\nabla_{\downarrow p,q} = I_{p,q} - I_{p,q+1} \tag{1}$$

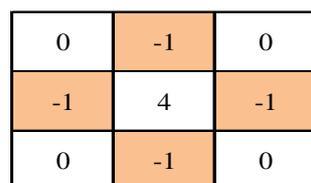
Similarly, first order pixel difference arrays in horizontal, main diagonal, and minor diagonal directions can be calculated as per the Figure 5a.



(a) first order pixel difference



(b) first separated pixel difference



(c) negative Laplacian operator

Figure 5. Pixel differences: (a) first order pixel difference; (b) separated pixel difference; (c) negative Laplacian operator.

Based on the aforementioned discussion in the Section 2, it is found that the incorporation of second and higher order pixel difference arrays has resulted in better accuracy as additional information is extracted by the incorporation of second and higher order pixel difference arrays. However, there are some shortcomings of higher order pixel difference arrays that are discussed in forthcoming contents.

The second order pixel difference array contains from range  $-510$  to  $510$  in the case of 8-bit image. The thresholding loose more information in the second order pixel difference array in comparison to the first order pixel difference array because the range of elements is higher than the first order. Therefore, separated pixel difference array, which provides additional information, and with intact range of elements,  $-255$  to  $255$ , are being used instead of the second order pixel difference array. For an image  $I$ , the separated pixel difference array in vertical direction can be represented as follows:

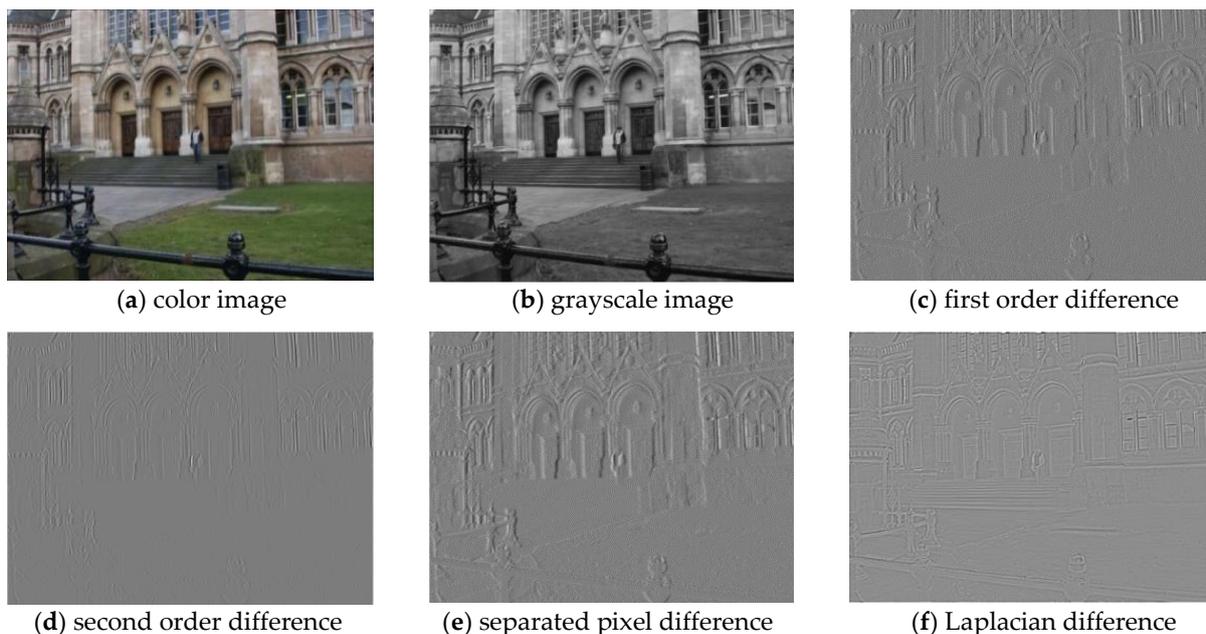
$$\nabla_{p,q}^s = I_{p,q} - I_{p+2,q} \quad (2)$$

Likewise, the separated pixel difference arrays in horizontal, main diagonal, and minor diagonal directions can be calculated as shown in the Figure 5b.

In addition to the first order and separated pixel difference arrays, Laplacian operator can also provide more suitable statistical information as it highlights abrupt changes clearly and provides the combined statistical information of horizontal and vertical neighbors. So, the proposed detector makes use of negative Laplacian operator to get the Laplacian difference array. The Laplacian operator can be defined as shown in Figure 5c:

$$\nabla_{p,q}^L = 4 \times I_{p,q} - (I_{p+1,q} + I_{p-1,q} + I_{p,q+1} + I_{p,q-1}) \quad (3)$$

The image and its corresponding difference arrays—the first order, the second order in the vertical direction, the separated pixel and Laplacian—are all shown in Figure 6. It is evident that each difference array provides the constructive statistical information of the image. Therefore, the features are extracted from the first order pixel difference arrays, separated pixel difference arrays in vertical, horizontal, diagonal, minor diagonal directions and Laplacian difference arrays in the low bit image, which helps in achieving better performance.



**Figure 6.** Image and their corresponding difference arrays: (a) color image; (b) grayscale image; (c) first order difference array; (d) second order difference array; (e) separated pixel difference array; (f) Laplacian difference array.

Next, the second order Markov chain is extracted to identify the co-occurrences of difference pixel arrays. Before understanding the second order Markov chain, we have to understand the first order Markov chain firstly. The first order Markov chain for the thresholded vertical first order pixel difference array ( $\nabla \downarrow$ ) can be defined as follows:

$$\chi_{\alpha,\beta} = Pr(\nabla \downarrow_{p,q+1} = \alpha \mid \nabla \downarrow_{p,q} = \beta) \quad (4)$$

where  $\chi_{\alpha,\beta} = 0$  for  $Pr(\nabla \downarrow_{p,q} = \beta) = 0$ . The second order Markov model can be defined for the thresholded vertical first order pixel difference array ( $\nabla \downarrow$ ):

$$\chi_{\alpha,\beta,\gamma} = Pr(\nabla \downarrow_{p,q+2} = \alpha \mid \nabla \downarrow_{p,q+1} = \beta, \nabla \downarrow_{p,q} = \gamma) \quad (5)$$

where  $\alpha, \beta, \gamma \in \{-\Gamma, -\Gamma + 1, \dots, 0, \dots, \Gamma - 1, \Gamma\}$  and  $\chi_{\alpha,\beta,\gamma} = 0$  for  $Pr(\nabla \downarrow_{p,q+1} = \beta, \nabla \downarrow_{p,q} = \gamma) = 0$  in addition.

Thus, nine feature vectors ( $\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6, \Omega_7, \Omega_8, \Omega_9$ ) are fetched using the Markov chain corresponding to the first order and separated pixel difference arrays in vertical, horizontal, diagonal and minor diagonal directions and the Laplacian difference array. The average of vertical and horizontal difference arrays feature vectors is considered. Similarly, the average value is taken for diagonal and minor diagonal feature vectors. The average is considered to normalize the varying characteristics to the flipping and mirroring operations. Finally, the final feature vector can be defined as follows:

$$\Omega = \frac{(\Omega_1 + \Omega_2)}{2} + \frac{(\Omega_3 + \Omega_4)}{2} + \frac{(\Omega_5 + \Omega_6)}{2} + \frac{(\Omega_7 + \Omega_8)}{2} + \Omega_9 \quad (6)$$

The dimension of resultant feature array can be obtained by following equation:

$$|\Omega| = \frac{(|\Omega_1| + |\Omega_2|)}{2} + \frac{(|\Omega_3| + |\Omega_4|)}{2} + \frac{(|\Omega_5| + |\Omega_6|)}{2} + \frac{(|\Omega_7| + |\Omega_8|)}{2} + |\Omega_9| \quad (7)$$

$$|\Omega| = 3 \times (2 \times \Gamma + 1)^3 \quad (8)$$

The second order Markov chain generates  $(2 \times \Gamma + 1)^3$  dimension array for the threshold ( $\Gamma$ ). The detailed experimental analysis is performed to detect median, mean and Gaussian filtering on UCID database in the subsequent section.

#### 4. Experimental Results

This section demonstrates the efficacy of the proposed method by carrying out an experimental study. The UCIDv2.0 [36] database (UCID) has been selected to evaluate the performance of the proposed method since the UCID database has been one of the prime databases being used in literature. All experimental results are shown using UCID database in Tables 1–6 excepting Table 4, where multiple databases are used for experimental analysis. UCID contains 1338 images of size  $384 \times 512$  and  $512 \times 384$  pixels. However, the images of size  $256 \times 256$ ,  $128 \times 128$ ,  $64 \times 64$ , and  $32 \times 32$  pixels are considered for experimental purposes by cropping center part of original images of UCID database images as shown in Figure 7b, which can eliminate the padding artifacts. In particular, zero padding artifacts are more of a serious concern in filtering detection. The performance of the proposed method (HSB-SPAM) is compared with some of the popular and closely related image forensic methods such as GLF [12], GDCTF [30], PERB [17], and SPAM [10].

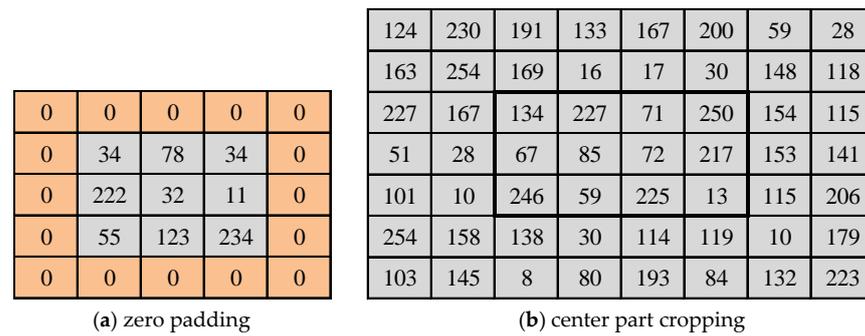


Figure 7. Padding and center cropping: (a) zero padding; (b) center part cropping.

The performance is evaluated by an average percentage accuracy on 100 training-testing pairs using two classifiers, namely support vector machines (SVM) and linear discriminant analysis (LDA). The analysis of experimental results is presented in four-subsections defined as follows.

Both uncompressed and compressed images with JPEG quality  $Q = \{70,50,30\}$  and size of  $128 \times 128$ ,  $64 \times 64$ , and  $32 \times 32$  pixels are used for experimental purposes. The non-filtered images are denoted by ORI. Median filtered images with a filter size of  $3 \times 3$  are denoted by MF3. Similarly, mean filtered and Gaussian filtered images of a filter size  $3 \times 3$  are denoted by AVG3 and GAU3, respectively. Median, mean and Gaussian filtered images of filter window size  $5 \times 5$  are denoted by MF5, AVG5 and GAU5, respectively. The compression is applied after filtering, i.e., post JPEG compression in filtered images. The SVM classifier with linear kernel is taken as a classifier. More than one hundred training testing pairs are formed for unbiased result analysis. Both the training and testing set have an equal number of images. The experimental results are represented in the form of average detection accuracy and average percentage error. For the proposed method, optimum SVM hyper-parameters are achieved using minimum classification error plot as shown in Figure 8. The estimated and observed minimum classification error point denotes the estimated and observed error while considering different hyper parameter values in the process of Bayesian optimization. The best hyper-parameter point denotes the iteration of corresponding optimized hyper-parameters. The minimum error hyper-parameters point defines the iteration of respective hyper parameters that produce the observed minimum classification error. The ORI and MF3 images of image sizes  $64 \times 64$  and JPEG  $Q = 50$  are taken for parameter selection. The hyper-parameters are utilized for all other experimental results in rest of the paper.

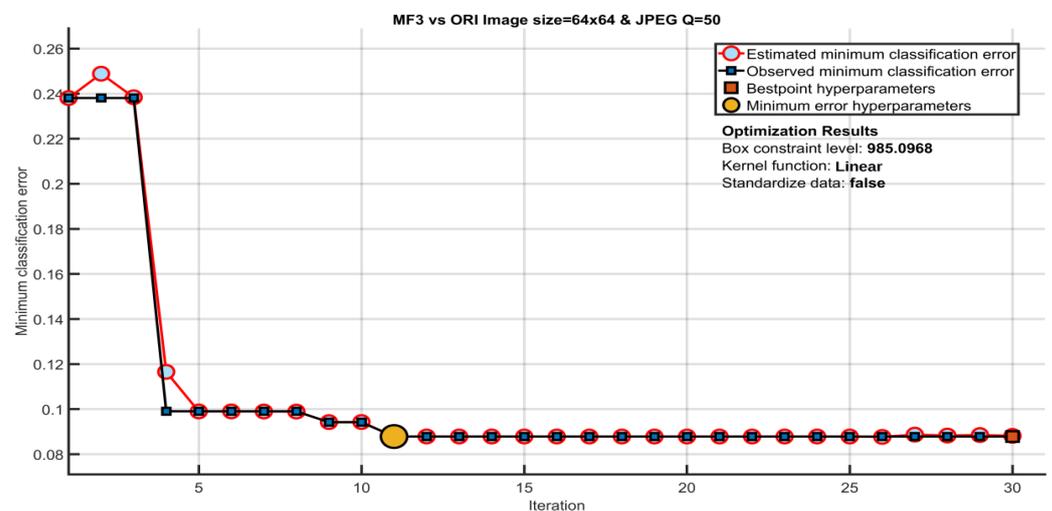


Figure 8. Optimization of the SVM hyper-parameters selection.

Experimental results are discussed in two sections. In Section 4.1, results are discussed between non-filtered and filtered images in Tables 1–3 on UCID dataset and in Table 4 for large dataset. In Section 4.2, Table 5 displayed the results where both sets contain only filtered images either of different filter size, different filter or both. In Table 6, different sized filtered images are considered in the training and testing set.

#### 4.1. Results for Non-Filtered and Filtered Images

In this section, results are displayed for non-filter vs. filtered images on different image size and compression quality. In Table 1, the percentage of detection error for ORI vs. MF3 and ORI vs. MF5 images is provided. The results of the proposed method (HSB-SPAM) are outstanding in the most of the cases as there is significant improvement on low resolution and compressed images. Best results of each case are highlighted by the shaded background in the tables. The detection of the  $3 \times 3$  filter size median filtered images is challenging than the  $5 \times 5$  filter size median filtered images. The performance of PERB and GDCTF methods are comparatively low. It shows that many existing techniques performance is good in the case of zero padded median filtered images only. Especially, the zero padding increases the occurrence of zero's in the filtered image in small size images (Figure 7a). The higher number of zero's helps the classifier to classify the images easily. To overcome the zero artifacts, the first filtering is applied on the large size image and then the required image size is cropped from the center region.

**Table 1.** Percentage of detection error on MF3 and MF5 images.

IMAGE SIZE	JPEG COM.	ORI vs. MF3					ORI vs. MF5				
		GLF	GDCTF	PERB	SPAM	HSB-SPAM	GLF	GDCTF	PERB	SPAM	HSB-SPAM
128 × 128	No	0.38	12.67	6.32	0.00	0.38	0.76	8.04	5.28	0.00	0.83
	Q = 70	6.56	16.08	9.10	7.08	3.40	2.74	9.92	6.53	3.02	2.33
	Q = 50	9.51	17.26	10.00	11.91	4.79	3.51	10.30	6.91	4.90	3.65
	Q = 30	12.08	19.41	11.46	15.80	7.19	4.79	10.81	7.60	7.53	4.38
64 × 64	No	0.59	17.49	9.41	0.17	0.83	1.35	10.24	7.99	0.17	1.39
	Q = 70	9.41	20.90	14.48	12.05	6.42	4.93	11.87	10.07	6.67	4.59
	Q = 50	13.30	21.97	14.79	17.19	8.68	6.39	13.29	10.63	10.24	5.94
	Q = 30	15.83	23.47	16.22	21.84	11.49	7.83	14.72	11.04	12.15	6.98
32 × 32	No	1.01	23.68	13.68	0.28	1.35	1.77	18.53	11.01	0.42	2.12
	Q = 70	14.93	25.66	19.97	19.51	10.73	7.92	20.13	13.75	11.46	8.40
	Q = 50	18.54	27.91	22.26	23.96	15.73	10.12	21.21	14.58	13.09	9.86
	Q = 30	21.98	29.09	23.65	28.58	19.41	11.91	22.11	15.45	17.67	11.67

In Table 2, experimental results are shown for detection of non-filtered and mean filtered images, ORI vs. AVG3 and ORI vs. AVG5. The results are given for image size  $128 \times 128$ ,  $64 \times 64$  and  $32 \times 32$ . The different image set are constructed using post JPEG compression of quality  $Q = \{70, 50, 30\}$ . The detection of mean filtered images of  $3 \times 3$  filter size is more challenging than the  $5 \times 5$  filter size. The performance of the proposed method is better than GLF technique. The performance of GDCTF, PERB and SPAM is inferior in the most of the cases. Actually, the detection of median filtered is more challenging than any other filter because median filter is a non-linear filter.

**Table 2.** Percentage of detection error on AVG3 and AVG5 images.

IMAGE SIZE	JPEG COM.	ORI vs. AVG3					ORI vs. AVG5				
		GLF	GDCTF	PERB	SPAM	HSB-SPAM	GLF	GDCTF	PERB	SPAM	HSB-SPAM
128 × 128	No	0.63	5.11	5.63	0.45	0.56	0.56	2.36	3.40	0.66	0.76
	Q = 70	1.46	4.35	6.11	2.92	0.90	1.15	3.99	3.85	1.39	0.76
	Q = 50	1.46	5.21	6.08	4.76	1.01	1.60	4.65	3.92	2.40	1.15
	Q = 30	1.81	6.22	6.88	6.39	0.97	1.67	4.97	3.65	2.60	1.56
64 × 64	No	1.39	8.53	8.75	1.91	1.49	1.32	6.01	5.73	1.53	0.69
	Q = 70	3.72	9.19	10.07	5.73	3.33	2.92	7.33	7.57	3.92	1.91
	Q = 50	4.93	10.31	10.49	7.99	4.41	3.19	8.02	7.67	5.49	2.19
	Q = 30	6.39	12.63	11.11	10.14	5.73	3.33	8.65	7.40	5.42	2.78
32 × 32	No	1.53	15.69	12.26	2.71	2.26	2.19	14.24	9.03	2.64	1.25
	Q = 70	6.15	18.53	14.62	9.79	5.18	4.76	15.73	11.22	7.12	4.10
	Q = 50	8.85	19.36	15.66	13.26	7.79	5.90	16.60	11.28	8.51	4.79
	Q = 30	11.18	20.03	15.97	15.69	9.83	6.18	18.06	10.73	9.83	5.59

The Gaussian filter is also used to hide the traces of forgery. From analysis of experimental results provided in Table 3, the detection of Gaussian filtered images is more challenging than the mean filtered images though less challenging than median filtered images. Even for small size,  $32 \times 32$  pixels and JPEG Q = 30, the proposed method gives more than 88% and 91% detection accuracy for GAU3 and GAU5, respectively. PERB method [8] give slightly better results for large size ( $128 \times 128$ ) and uncompressed image set for the Gaussian filter  $3 \times 3$ . However, the computation cost of PERB is the highest.

**Table 3.** Percentage of detection error on GAU3 and GAU5 images.

IMAGE SIZE	JPEG COM.	ORI vs. GAU3					ORI vs. GAU5				
		GLF	GDCTF	PERB	SPAM	HSB-SPAM	GLF	GDCTF	PERB	SPAM	HSB-SPAM
128 × 128	No	0.73	5.99	0.45	0.35	0.63	1.01	3.65	3.19	0.45	0.87
	Q = 70	2.19	8.28	1.39	3.47	1.63	2.33	5.01	4.44	1.98	1.91
	Q = 50	2.74	9.39	1.46	4.55	2.19	2.67	6.16	4.72	3.16	2.26
	Q = 30	3.13	10.30	1.56	5.45	2.47	3.09	6.93	5.03	3.82	2.50
64 × 64	No	1.49	9.43	7.60	0.69	1.56	1.25	5.99	5.80	0.73	1.18
	Q = 70	4.48	11.24	10.38	7.05	3.62	3.23	6.79	7.78	3.82	3.13
	Q = 50	6.53	13.36	11.04	8.99	5.66	4.44	7.83	8.58	5.90	3.89
	Q = 30	8.51	15.58	12.12	12.22	7.19	5.24	9.68	9.24	7.33	4.93
32 × 32	No	1.74	16.83	11.42	1.84	2.15	1.42	12.74	7.99	1.15	1.67
	Q = 70	7.81	17.81	14.93	11.91	7.47	5.07	13.34	11.74	7.81	4.90
	Q = 50	10.59	19.79	16.18	14.79	9.66	7.40	16.14	12.78	9.65	6.91
	Q = 30	13.72	22.91	17.53	19.72	11.95	9.79	18.40	13.72	13.23	8.96

A large dataset of ten thousand images is created by using UCID, BOWS2 [37], Columbia [38], and RAISE [39] datasets. The images of  $64 \times 64$  are created by cropping center part. In Table 6, percentage of detection error is displayed for both  $3 \times 3$  &  $5 \times 5$  filter of median, mean and Gaussian filtering. The results are given for non-compressed (NC), and compressed images with JPEG Q = {70,50,30}. However, the results are shown the same pattern as shown in Table 1 for small sized dataset.

**Table 4.** Percentage of detection error on  $64 \times 64$  filtered images.

	Technique	NC	Q = 70	Q = 50	Q = 30		Technique	NC	Q = 70	Q = 50	Q = 30
ORI vs. MF3	GLF	0.63	10.35	13.43	17.26	ORI vs. MF5	GLF	1.39	4.93	6.71	7.99
	GDCTF	19.24	22.78	24.17	23.93		GDCTF	11.06	12.11	13.56	15.16
	PERB	9.60	15.49	16.27	17.84		PERB	8.39	10.07	11.05	11.59
	SPAM	0.19	13.01	17.36	22.71		SPAM	1.19	7.13	11.06	12.52
	HSB-SPAM	0.88	6.49	9.46	11.61		HSB-SPAM	0.47	4.64	6.53	7.61
ORI vs. AVG3	GLF	1.43	3.90	5.03	6.96	ORI vs. AVG5	GLF	1.45	2.92	3.51	3.63
	GDCTF	8.87	9.19	10.92	13.01		GDCTF	6.55	7.47	8.34	9.51
	PERB	9.10	10.88	11.01	12.00		PERB	5.84	8.02	8.44	7.99
	SPAM	2.04	6.07	8.39	10.14		SPAM	1.59	4.20	5.49	5.58
	HSB-SPAM	1.57	3.37	4.76	5.90		HSB-SPAM	0.76	1.97	2.30	3.00
ORI vs. GAU3	GLF	1.64	4.75	6.92	8.93	ORI vs. GAU5	GLF	1.33	3.33	4.53	5.61
	GDCTF	9.43	11.69	14.30	16.36		GDCTF	6.41	7.27	8.46	9.87
	PERB	7.83	11.11	11.26	12.85		PERB	6.20	8.17	9.09	9.61
	SPAM	0.71	7.12	9.71	13.32		SPAM	0.79	4.20	6.32	7.47
	HSB-SPAM	1.56	3.62	5.66	7.62		HSB-SPAM	1.18	3.44	4.12	4.98

#### 4.2. Results for Complex Scenarios

In this section, two complex scenarios are considered using UCID dataset. In Table 5, the comparative results analysis between filtered images is provided. The provided results represent the percentage detection accuracy for the image set of size  $64 \times 64$  pixels. The image set contains uncompressed and post JPEG compressed image with quality  $Q = \{70, 50, 30\}$ . The accuracy between median filtered images and other filtered images are greater than 95% in uncompressed image set. The detection between filtered image set pairs, AVG3 vs. GAU3, AVG3 vs. GAU5 and GAU3 vs. GAU5 are affected by post JPEG compression and detection accuracy sharply decreased. In the case of JPEG compressed images, HSB-SPAM method outperforms the existing methods such as SPAM, GLF, GDCTF and PERB whereas the performance of all the methods including HSB-SPAM, SPAM and GLF is quite promising in the case of uncompressed images.

Table 5. Percentage of detection accuracy on filtered images.

METHODS	MF3	MF3	MF3	MF3	MF3	MF5	MF5	MF5	MF5	AVG3	AVG3	AVG5	AVG5	AVG3	GAU3
	MF5	AVG3	AVG5	GAU3	GAU5	AVG3	AVG5	GAU3	GAU5	GAU3	GAU5	GAU3	GAU5	AVG5	GAU5
Image Size $64 \times 64$ Uncompressed															
GLF	94.55	99.27	98.00	99.17	99.27	98.96	98.19	98.99	99.10	95.00	94.00	97.99	97.48	96.00	84.00
GDCTF	67.00	76.00	86.00	74.00	81.00	70.00	80.00	70.00	75.00	70.00	70.00	77.00	70.00	75.00	67.00
PERB	83.54	81.08	88.33	79.55	81.11	85.49	76.70	88.02	87.19	61.53	64.27	86.98	82.74	82.85	66.56
SPAM	98.96	99.26	99.48	99.69	99.69	99.72	99.65	99.72	99.76	87.88	92.53	96.77	95.80	87.15	84.03
HSB-SPAM	96.32	99.82	99.65	99.41	99.48	99.34	98.30	99.41	99.55	96.90	96.00	98.33	97.99	97.78	86.00
Image Size $64 \times 64$ JPEG Q = 70															
GLF	89.93	93.58	96.76	86.67	92.78	95.38	93.85	93.72	94.97	82.57	69.55	97.35	96.15	96.56	76.70
GDCTF	68.00	76.00	87.00	70.00	80.00	66.00	79.00	80.00	69.00	57.00	58.00	91.00	67.00	73.00	62.00
PERB	81.91	74.86	88.58	69.20	81.22	85.31	80.87	87.47	88.68	58.82	56.81	87.33	81.46	82.88	67.47
SPAM	83.40	87.05	93.75	76.91	90.97	91.81	90.52	91.18	92.53	59.65	61.11	95.63	84.20	86.22	74.27
HSB-SPAM	91.99	96.38	98.74	92.60	96.52	98.29	96.87	95.69	97.74	82.81	71.93	99.13	98.53	98.56	78.95
Image Size $64 \times 64$ JPEG Q = 50															
GLF	87.57	90.49	97.71	87.74	90.97	93.19	93.30	92.85	92.64	75.83	62.33	96.25	95.31	96.53	74.31
GDCTF	67.00	74.00	87.00	70.00	79.00	64.00	77.00	63.00	66.00	60.00	59.00	79.00	69.00	73.00	63.00
PERB	81.39	75.87	89.41	68.47	82.67	84.90	79.51	86.15	86.42	60.03	55.63	87.60	81.08	83.02	67.57
SPAM	79.65	83.44	92.36	78.33	87.92	87.29	88.16	86.98	87.12	59.34	57.12	89.20	83.16	84.13	70.31
HSB-SPAM	89.68	95.06	98.43	92.18	95.65	96.10	95.76	95.06	95.65	78.00	64.89	98.50	97.34	98.00	76.63
Image Size $64 \times 64$ JPEG Q = 30															
GLF	86.42	87.74	97.15	80.73	89.10	88.75	92.53	87.71	87.33	71.77	58.99	96.15	94.24	95.38	71.94
GDCTF	67.00	71.00	87.00	65.00	75.00	61.00	76.00	61.00	64.00	60.00	52.99	80.00	71.00	75.00	64.00
PERB	81.04	75.38	89.31	67.53	82.05	81.11	79.83	82.88	82.36	59.31	55.07	87.60	81.18	82.85	69.65
SPAM	76.77	79.34	92.29	69.69	84.27	79.51	85.31	80.17	79.34	54.17	54.51	88.82	81.91	84.03	68.37
HSB-SPAM	88.35	92.11	97.88	88.05	93.64	92.56	94.85	91.80	91.97	71.42	61.24	97.43	96.13	96.07	75.03

In Table 6, the different filter size is used in training and testing set. For example, if ORI and MF3 pair is used in training set, then ORI and MF5 pair is used in testing, where median, mean and Gaussian filtering with filter size  $3 \times 3$  and  $5 \times 5$  is used. In experimental results, the proposed method are better results for  $3 \times 3$  filter size in training pair and  $5 \times 5$  in testing pair in comparison to  $5 \times 5$  filter size in training pair and  $3 \times 3$  in testing pair. In most of the cases, the detection error of the proposed method is less than other methods.

**Table 6.** Percentage of detection error for different filter size on training and testing sets.

JPEG COM.	Training: ORI-MF3 & Testing: ORI-MF5					Training: ORI-MF5 & Testing: ORI-MF3				
	GLF	GDCTF	PERB	SPAM	HSB- SPAM	GLF	GDCTF	PERB	SPAM	HSB- SPAM
NC	0.97	17.49	10.76	1.08	0.07	15.84	22.87	20.59	0.11	1.38
Q = 70	9.68	21.00	14.24	8.33	5.83	34.68	24.81	26.05	25.69	19.02
Q = 50	12.37	21.94	14.84	11.73	7.92	35.46	27.80	28.21	27.22	26.42
Q = 30	13.64	23.36	15.96	12.89	10.20	34.79	30.04	29.37	30.23	28.55
	Training: ORI-AVG3 & Testing: ORI-AVG5					Training: ORI-AVG5 & Testing: ORI-AVG3				
	GLF	GDCTF	PERB	SPAM	HSB- SPAM	GLF	GDCTF	PERB	SPAM	HSB- SPAM
NC	11.51	8.41	8.26	1.27	3.14	18.54	15.88	16.89	16.48	17.00
Q = 70	19.02	10.16	9.30	5.31	4.07	19.99	19.21	18.87	29.93	16.93
Q = 50	11.29	11.10	9.90	5.49	5.08	20.18	20.37	20.59	32.10	18.31
Q = 30	8.63	12.71	10.01	6.32	4.97	20.18	21.00	21.79	36.25	21.11
	Training: ORI-GAU3 & Testing: ORI-GAU5					Training: ORI-GAU5 & Testing: ORI-GAU3				
	GLF	GDCTF	PERB	SPAM	HSB- SPAM	GLF	GDCTF	PERB	SPAM	HSB- SPAM
NC	1.08	9.12	7.62	0.89	0.71	1.61	14.20	9.94	1.42	1.27
Q = 70	3.59	11.62	9.94	3.96	3.33	6.09	16.59	11.40	8.26	6.24
Q = 50	5.23	13.23	10.05	5.57	3.92	7.92	18.12	11.70	10.13	7.88
Q = 30	6.13	15.10	11.81	7.21	4.56	10.46	21.00	12.97	13.15	8.48

## 5. Conclusions

In this paper, a filtering detection method based on HSB-SPAM has been proposed without any prior information on the modification or forgery. The performance of the proposed detector was verified to detect three types of filtering operations—Median, mean and Gaussian filtering. The proposed method gave higher accuracy than the conventional detectors, especially on JPEG compressed images. The main reason behind the higher accuracy was the use of higher significant bit-plane for difference arrays, where more effective feature set could be constructed by the proposed method. In addition, the proposed method considered the multiple difference arrays for stronger feature set. As a result, the proposed detector with the Markov chain could achieve a higher accuracy than the existing filtering detection techniques in the most cases of the scenarios. Further, the proposed method was also superior in terms of cross-analysis. In the future, the proposed method will be examined for some other types of image operators that can provide promising results in many scenarios and robust techniques can also be proposed that can detect modifications of the image.

**Author Contributions:** Both authors discussed the contents of the manuscript and contributed equally to its preparation. S.A. designed and implemented the proposed technique. K.-H.J. designed the manuscript and conducted the literature review. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by Brain Pool program funded by the Ministry of Science and ICT through the National Research Foundation of Korea (2019H1D3A1A01101687) and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2018R1D1A1A09081842).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The datasets used in this paper are publically available and their links are provided in the reference section.

**Acknowledgments:** We thank the anonymous reviewers for their valuable suggestions that improved the quality of this article.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Matern, F.; Riess, C.; Stamminger, M. Gradient-Based Illumination Description for Image Forgery Detection. *IEEE Trans. Inf. Forensics Secur.* **2020**, *15*, 1303–1317. [[CrossRef](#)]
2. Moghaddasi, Z.; Jalab, H.A.; Noor, R.M. Image splicing forgery detection based on low-dimensional singular value decomposition of discrete cosine transform coefficients. *Neural Comput. Appl.* **2019**, *31*, 7867–7877. [[CrossRef](#)]
3. Marra, F.; Gragnaniello, D.; Verdoliva, L.; Poggi, G. A Full-Image Full-Resolution End-to-End-Trainable CNN Framework for Image Forgery Detection. *IEEE Access* **2020**, *8*, 133488–133502. [[CrossRef](#)]
4. Almabdy, S.; Elrefaei, L. Deep convolutional neural network-based approaches for face recognition. *Appl. Sci.* **2019**, *9*, 4397. [[CrossRef](#)]
5. Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep Learning for Generic Object Detection: A Survey. *Int. J. Comput. Vis.* **2020**, *128*, 261–318. [[CrossRef](#)]
6. Paszkiel, S. Using Neural Networks for Classification of the Changes in the EEG Signal Based on Facial Expressions. In *Studies in Computational Intelligence*; Springer: Cham, Switzerland, 2020; Volume 852, pp. 41–69.
7. Marra, F.; Poggi, G.; Roli, F.; Sansone, C.; Verdoliva, L. Counter-forensics in machine learning based forgery detection. In Proceedings of the Media Watermarking, Security, and Forensics, San Francisco, CA, USA, 9–11 February 2015; Alattar, A.M., Memon, N.D., Heitzenrater, C.D., Eds.; SPIE: Bellingham, WA, USA, 2015; Volume 9409, p. 94090L.
8. Kim, D.; Jang, H.-U.; Mun, S.-M.; Choi, S.; Lee, H.-K. Median Filtered Image Restoration and Anti-Forensics Using Adversarial Networks. *IEEE Signal Process. Lett.* **2018**, *25*, 278–282. [[CrossRef](#)]
9. Gragnaniello, D.; Marra, F.; Poggi, G.; Verdoliva, L. Analysis of Adversarial Attacks against CNN-based Image Forgery Detectors. In Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), Rome, Italy, 3–7 September 2018; IEEE: New York, NY, USA, 2018; Volume 2018, pp. 967–971.
10. Kirchner, M.; Fridrich, J. On detection of median filtering in digital images. In Proceedings of the Media Forensics and Security II, San Jose, CA, USA, 27 January 2010; Memon, N.D., Dittmann, J., Alattar, A.M., Delp, E.J., III, Eds.; SPIE: Bellingham, WA, USA, 2010; Volume 7541, p. 754110.
11. Pevny, T.; Bas, P.; Fridrich, J. Steganalysis by Subtractive Pixel Adjacency Matrix. *IEEE Trans. Inf. Forensics Secur.* **2010**, *5*, 215–224. [[CrossRef](#)]
12. Chen, C.; Ni, J.; Huang, J. Blind Detection of Median Filtering in Digital Images: A Difference Domain Based Approach. *IEEE Trans. Image Process.* **2013**, *22*, 4699–4710. [[CrossRef](#)]
13. Cao, G.; Zhao, Y.; Ni, R.; Yu, L.; Tian, H. Forensic detection of median filtering in digital images. In Proceedings of the 2010 IEEE International Conference on Multimedia and Expo, Singapore, 19–23 July 2010; pp. 89–94. [[CrossRef](#)]
14. Agarwal, S.; Chand, S.; Skarbnik, N. SPAM revisited for median filtering detection using higher-order difference. *Secur. Commun. Netw.* **2016**, *9*, 4089–4102. [[CrossRef](#)]
15. Niu, Y.; Zhao, Y.; Ni, R.R. Robust median filtering detection based on local difference descriptor. *Signal Process. Image Commun.* **2017**, *53*, 65–72. [[CrossRef](#)]
16. Kang, X.; Stamm, M.C.; Peng, A.; Liu, K.J.R. Robust median filtering forensics using an autoregressive model. *IEEE Trans. Inf. Forensics Secur.* **2013**, *8*, 1456–1468. [[CrossRef](#)]
17. Ferreira, A.; Rocha, A. A Multiscale and Multi-Perturbation Blind Forensic Technique for Median Detecting. In *Iberoamerican Congress on Pattern Recognition*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2014; Volume 8827, pp. 302–310. ISBN 9783319125671.
18. Chen, J.; Kang, X.; Liu, Y.; Wang, Z.J. Median Filtering Forensics Based on Convolutional Neural Networks. *IEEE Signal Process. Lett.* **2015**, *22*, 1849–1853. [[CrossRef](#)]
19. Hwang, J.J.; Rhee, K.H. Gaussian filtering detection based on features of residuals in image forensics. In Proceedings of the 2016 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), Hanoi, Vietnam, 7–9 November 2016; IEEE: New York, NY, USA, 2016; pp. 153–157.

20. Liu, A.; Zhao, Z.; Zhang, C.; Su, Y. Median filtering forensics in digital images based on frequency-domain features. *Multimed. Tools Appl.* **2017**, *76*, 22119–22132. [[CrossRef](#)]
21. Yang, J.; Ren, H.; Zhu, G.; Shi, J.H.Y. Detecting median filtering via two-dimensional AR models of multiple filtered residuals. *Multimed. Tools Appl.* **2017**, *77*, 7931–7953. [[CrossRef](#)]
22. Tang, H.; Ni, R.; Zhao, Y.; Li, X. Median filtering detection of small-size image based on CNN. *J. Vis. Commun. Image Represent.* **2018**, *51*, 162–168. [[CrossRef](#)]
23. Bayar, B.; Stamm, M.C. Constrained Convolutional Neural Networks: A New Approach Towards General Purpose Image Manipulation Detection. *IEEE Trans. Inf. Forensics Secur.* **2018**, *13*, 2691–2706. [[CrossRef](#)]
24. Wang, D.; Gao, T. Filtered Image Forensics Based on Frequency Domain Features. In Proceedings of the 2018 IEEE 18th International Conference on Communication Technology (ICCT), Chongqing, China, 8–11 October 2018; IEEE: New York, NY, USA, 2018; Volume 2019, pp. 1208–1212.
25. Li, W.; Ni, R.; Li, X.; Zhao, Y. Robust median filtering detection based on the difference of frequency residuals. *Multimed. Tools Appl.* **2019**, *78*, 8363–8381. [[CrossRef](#)]
26. Peng, A.; Luo, S.; Zeng, H.; Wu, Y. Median filtering forensics using multiple models in residual domain. *IEEE Access* **2019**, *7*, 28525–28538. [[CrossRef](#)]
27. Rhee, K.H. Improvement Feature Vector: Autoregressive Model of Median Filter Residual. *IEEE Access* **2019**, *7*, 77524–77540. [[CrossRef](#)]
28. Rhee, K.H. Forensic Detection Using Bit-Planes Slicing of Median Filtering Image. *IEEE Access* **2019**, *7*, 92586–92597. [[CrossRef](#)]
29. Luo, S.; Peng, A.; Zeng, H.; Kang, X.; Liu, L. Deep Residual Learning Using Data Augmentation for Median Filtering Forensics of Digital Images. *IEEE Access* **2019**, *7*, 80614–80621. [[CrossRef](#)]
30. Gupta, A.; Singhal, D. A simplistic global median filtering forensics based on frequency domain analysis of image residuals. *ACM Trans. Multimed. Comput. Commun. Appl.* **2019**, *15*, 1–23. [[CrossRef](#)]
31. Peng, A.; Yu, G.; Wu, Y.; Zhang, Q.; Kang, X. A universal image forensics of smoothing filtering. *Int. J. Digit. Crime Forensics* **2019**, *11*, 18–28. [[CrossRef](#)]
32. Hwang, J.J.; Rhee, K.H. Gaussian Forensic Detection using Blur Quantity of Forgery Image. In Proceedings of the 2019 International Conference on Green and Human Information Technology (ICGHIT), Kuala Lumpur, Malaysia, 16–18 January 2019; IEEE: New York, NY, USA, 2019; pp. 86–88.
33. Gao, H.; Gao, T. Detection of median filtering based on ARMA model and pixel-pair histogram feature of difference image. *Multimed. Tools Appl.* **2020**, *79*, 12551–12567. [[CrossRef](#)]
34. Gao, H.; Gao, T.; Cheng, R. Robust detection of median filtering based on data-pair histogram feature and local configuration pattern. *J. Inf. Secur. Appl.* **2020**, *53*, 102506. [[CrossRef](#)]
35. Zhang, J.; Liao, Y.; Zhu, X.; Wang, H.; Ding, J. A Deep Learning Approach in the Discrete Cosine Transform Domain to Median Filtering Forensics. *IEEE Signal Process. Lett.* **2020**, *27*, 276–280. [[CrossRef](#)]
36. Schaefer, G.; Stich, M. UCID: An uncompressed color image database. In Proceedings of the Storage and Retrieval Methods and Applications for Multimedia 2004, San Jose, CA, USA, 20 January 2004; Yeung, M.M., Lienhart, R.W., Li, C.-S., Eds.; SPIE: Bellingham, WA, USA, 2003; Volume 5307, pp. 472–480.
37. Bas, P.; Furon, T. Break Our Watermarking System, 2nd ed. 2008. Available online: <http://bows2.ec-lille.fr/> (accessed on 22 June 2020).
38. Content, I.; Conventions, N. Columbia Uncompressed Image Splicing Detection Evaluation Dataset. 2014, pp. 2–5. Available online: <https://www.ee.columbia.edu/ln/dvmm/downloads/authspluncmp/> (accessed on 18 June 2020).
39. Dang-Nguyen, D.T.; Pasquini, C.; Conotter, V.; Boato, G. RAISE—A raw images dataset for digital image forensics. In Proceedings of the 6th ACM Multimedia Systems Conference, MMSys 2015, Portland, OR, USA, 18–20 March 2015; ACM: New York, NY, USA, 2015; pp. 219–224.